



TESIS TE142599

**DESAIN *FUZZY DISTURBANCE OBSERVER* DENGAN
NEURO-FUZZY INVERS MODEL UNTUK
PENGENDALIAN KECEPATAN *PARALLEL HYBRID*
*ELECTRIC VEHICLE***

YOGA ALIF KURNIA UTAMA
2213202015

DOSEN PEMBIMBING
Dr. Trihastuti Agustinah, ST., MT.
Ir. Josaphat Pramudijanto, M.Eng.

PROGRAM MAGISTER
BIDANG KEAHLIAN TEKNIK SISTEM PENGATURAN
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2015



TESIS TE142599

DESIGN FUZZY DISTURBANCE OBSERVER WITH NEURO-FUZZY INVERS MODEL FOR PARALLEL HYBRID ELECTRIC VEHICLE SPEED CONTROLLER

YOGA ALIF KURNIA UTAMA
2213202015

SUPERVISOR
Dr. Trihastuti Agustinah, ST., MT.
Ir. Josaphat Pramudijanto, M.Eng.

MAGISTER PROGRAM
CONTROL SYSTEM ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY OF INDUSTRIAL ENGINEERING
INSTITUTE OF TECHNOLOGY SEPULUH NOPEMBER
SURABAYA
2015

**Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (M.T)**

di

Institut Teknologi Sepuluh Nopember

Oleh:

**Yoga Alif Kurnia Utama
NRP. 2213202015**

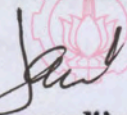
**Tanggal Ujian : 28 Juli 2015
Periode Wisuda : September 2015**

Disetujui oleh:



**1. Dr. Trihastuti Agustinah, ST., MT.
NIP. 1968 0812 1994 0320 01**

(Pembimbing I)



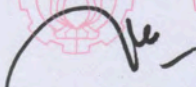
**2. Ir. Josaphat Pramudijanto, M.Eng.
NIP. 1962 1005 1990 0310 03**

(Pembimbing II)



**3. Ir. Rusdhianto Effendie, A.K., MT.
NIP. 1957 0424 1985 0210 01**

(Penguji I)



**4. Ir. Joko Susila, MT.
NIP. 1966 0606 1991 0210 01**

(Penguji II)



**5. Eka Iskandar, ST., MT.
NIP. 1980 0528 2008 1210 01**

(Penguji III)

Direktur Program Pascasarjana,



**Prof. Dr. Ir. Adi Soeprijanto, M.T.
NIP. 1964 0405 1990 0210 01**

DESAIN *FUZZY DISTURBANCE OBSERVER* DENGAN *NEURO-FUZZY INVERS MODEL* UNTUK PENGENDALIAN KECEPATAN *PARALLEL HYBRID ELECTRIC VEHICLE*

Nama Mahasiswa : Yoga Alif Kurnia Utama
NRP : 2213202015
Pembimbing : 1. Dr. Trihastuti Agustinah, S.T., M.T.
2. Ir. Josaphat Pramudijanto, M.Eng.

ABSTRAK

Pada dekade terakhir, terdapat banyak kendaraan transportasi yang menggunakan *Internal Combustion Engine* (ICE). Hal ini menimbulkan dampak serius pada lingkungan karena emisi gas polutan yang dikeluarkannya. Salah satu solusi adalah penggunaan *hybrid electric vehicle* (HEV) sebagai pengganti dari kendaraan yang menggunakan ICE. Salah satu performa yang harus dimiliki oleh HEV adalah memiliki kecepatan yang tetap saat melaju di atas jalan raya. Jalan raya yang tidak rata akan menyebabkan gangguan pada HEV sehingga kecepatannya menjadi tidak tetap. Pada penelitian ini, metode *fuzzy disturbance observer* digunakan untuk menolak gangguan yang diterima HEV yang dikarenakan oleh jalan yang tidak rata. *Neuro-fuzzy invers model* digunakan sebagai komponen *invers model* dalam *disturbance observer* dimana memiliki kelebihan dalam menggambarkan *invers model* pada *plant* tanpa membuat model matematikanya. Pengujian dilakukan dengan membandingkan antara dua metode kontrol yaitu *fuzzy* dan *fuzzy disturbance observer* pada enam tipe model jalan yang berbeda. Hasil simulasi menunjukkan bahwa metode *fuzzy disturbance observer* dengan *neuro-fuzzy invers model* dapat menolak *disturbance* yang dikarenakan oleh jalan raya yang tidak rata dan dapat mempertahankan kecepatannya.

Kata Kunci: *Parallel Hybrid Electric Vehicle, Fuzzy Control, Disturbance Observer, Neuro-Fuzzy Invers Model*

DESIGN FUZZY DISTURBANCE OBSERVER WITH NEURO-FUZZY INVERS MODEL FOR PARALLEL HYBRID ELECTRIC VEHICLE SPEED CONTROLLER

By : Yoga Alif Kurnia Utama
Student Identity Number : 2213202015
Supervisor : 1. Dr. Trihastuti Agustinah, S.T., M.T.
2. Ir. Josaphat Pramudijanto, M.Eng.

ABSTRACT

In the last decade, there are a lot of transportation vehicles that use Internal Combustion Engine (ICE). They make serious problem to natural environment that caused by gas emissions pollutant which produces from them. One of many solutions is using Hybrid Electric Vehicle (HEV) as replacement of ICE system in the vehicles. One of the performance that must be have in HEV is having a steady speed while driving down on the road. Bumpy road would cause a disturbance in HEV so that the speed of HEV will not steady. In this research, fuzzy disturbance observer method use to reject the disturbance of HEV that caused by bumpy road. Neuro-fuzzy inverse model use as a component of the inverse model in disturbance observer which has advantages that can describe the inverse model without make a mathematic model. Testing will compare between two controllers, those are fuzzy and fuzzy disturbance observer in six kind of road models. The simulation results show that the method of fuzzy disturbance observer with neuro-fuzzy inverse model can reject the disturbance that caused by bumpy road and maintain its speed.

Keyword: *Parallel Hybrid Electric Vehicle, Fuzzy Control, Disturbance Observer, Neuro-Fuzzy Invers Model*

KATA PENGANTAR

Pertama-tama penulis memanjatkan puji syukur atas kehadiran Allah SWT, karena atas ijin dan rahmat-Nya akhirnya penulis dapat menyelesaikan tesis dengan judul:

“Desain *Fuzzy Disturbance Observer* dengan *Neuro-Fuzzy Invers Model* untuk Pengendalian Kecepatan *Parallel Hybrid Electric Vehicle*”

Tesis ini disusun untuk memenuhi persyaratan menyelesaikan pendidikan dan untuk mendapatkan gelar Magister Teknik (M.T.) pada Bidang Keahlian Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari bahwa tanpa bantuan dan dukungan dari banyak pihak, tesis ini tidak dapat terselesaikan dengan baik. Oleh karena itu, pada kesempatan ini, penulis menyampaikan ucapan terima kasih kepada:

1. Ibu Dr. Trihastuti Agustinah, S.T., M.T., selaku dosen pembimbing pertama dan Bapak Ir. Josaphat Pramudijanto, M.Eng., selaku dosen pembimbing kedua yang dengan sabar membimbing dan membantu menghadapi permasalahan dalam penyelesaian tesis ini.
2. Seluruh staf pengajar Bidang Keahlian Teknik Sistem Pengaturan atas segala bantuannya dalam memberikan pemahaman yang berhubungan dengan penyelesaian tesis ini.
3. Kedua orang tua, yang selalu mendoakan dan memberi dukungan, yang tanpa keberadaan mereka, penulis tidak akan seperti saat ini.
4. Rekan mahasiswa S2 Bidang Keahlian Teknik Sistem Pengaturan yang selalu memberikan dukungan dan bantuan dalam proses penyelesaian tesis ini.

Penulis menyadari bahwa hasil dari tesis ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran akan sangat diharapkan, sehingga bisa digunakan dalam memperbaiki segala kekurangan yang ada di dalam tesis ini. Akhir kata, penulis berharap agar tesis ini dapat bermanfaat bagi kemajuan dan perkembangan ilmu pengetahuan di bidang teknologi dan pengaturan.

Surabaya, September 2015

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TESIS	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Kontribusi	3
1.6 Metodologi.....	4
BAB II KAJIAN PUSTAKA & LANDASAN TEORI	5
2.1 Kajian pustaka.....	5
2.2 Landasan Teori.....	7
2.2.1 <i>Parallel Hybrid Electric Vehicle</i> (HEV).....	8
2.2.1 <i>Disturbance Obsever</i>	21
2.2.1 <i>Fuzzy Logic Controller</i>	23
2.2.1 <i>Neuro-Fuzzy</i>	25
BAB III PERANCANGAN SISTEM	29
3.1 Gambaran Umum Sistem.....	29
3.2 Perancangan <i>Disturbance</i>	29

3.3 Perancangan <i>Fuzzy Controller</i>	33
3.4 Perancangan <i>Disturbance Observer</i>	38
3.5 Perancangan <i>Neuro-Fuzzy Invers Model</i>	40
BAB IV HASIL DAN PEMBAHASAN	45
4.1 Gambaran Umum Pengujian Sistem	45
4.2 Pengujian tanpa <i>Disturbance</i>	45
4.3 Pengujian dengan <i>Disturbance I</i>	46
4.4 Pengujian dengan <i>Disturbance II</i>	47
4.5 Pengujian dengan <i>Disturbance III</i>	49
4.6 Pengujian dengan <i>Disturbance IV</i>	50
4.7 Pengujian dengan <i>Disturbance V</i>	51
4.8 Pengujian dengan <i>Disturbance VI</i>	53
BAB V PENUTUP	55
5.1 Kesimpulan	55
5.2 Saran	56
DAFTAR PUSTAKA	57
BIOGRAFI PENULIS	59

DAFTAR TABEL

Tabel 2.1 Perbandingan Koefisien pada 2 Jenis <i>Engine</i>	16
Tabel 3.1 Model Jalan	32
Tabel 3.2 <i>Rule Base Fuzzy Controller</i>	37
Tabel 3.3 <i>Premise Parameters</i>	42
Tabel 3.4 <i>Consequent Parameters</i>	43
Tabel 4.1 <i>Steady State Error</i>	54

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Saat ini transportasi darat, dipenuhi oleh banyak kendaraan konvensional yang berbahan bakar fosil. Kendaraan yang berbahan bakar fosil ini, menggunakan tenaga penggerak berupa mesin pembakaran dalam atau biasa disebut *Internal Combustion Engine* (ICE). Salah satu kekurangan terbesar dari kendaraan yang menggunakan ICE adalah emisi polutan yang dihasilkan oleh kendaraan tersebut. Emisi polutan inilah sebagai salah satu penyebab pemanasan global atau *global warming* yang terjadi pada saat ini.

Penggunaan *hybrid electric vehicle* (HEV) merupakan salah satu solusi untuk mengatasi emisi polutan yang berasal dari kendaraan yang menggunakan ICE. HEV merupakan tipe kendaraan yang memiliki sumber penggerak ganda, yaitu ICE dan motor listrik. Sumber energinya berasal dari bahan bakar bensin dan energi listrik yang disimpan di baterai. Dengan adanya *hybrid engine*, diharapkan mampu mengurangi emisi gas buang, sehingga polusi yang dihasilkan jauh lebih kecil dari kendaraan konvensional.

Salah satu performa yang harus dimiliki oleh HEV adalah tahan terhadap gangguan yang ada di jalan raya. Gangguan ini bisa disebabkan oleh banyak hal, yaitu jalan yang tidak rata, angin, dan sebagainya. Gangguan-gangguan ini menyebabkan kecepatan kendaraan menjadi tidak stabil. Oleh karena itu, gangguan-gangguan tersebut harus diatasi sehingga kecepatan kendaraan menjadi lebih stabil.

Beberapa peneliti telah membuat penelitian mengenai pengaturan kecepatan pada HEV ini. Pada salah satu penelitian [1], pengujian dilakukan dengan menggunakan *fuzzy controller* pada simulasi kecepatan HEV berjenis HEV paralel dengan perubahan kecepatan referensi. Hasil penelitian ini menunjukkan bahwa, kecepatan aktual mengikuti kecepatan referensi sepanjang tidak ada perubahan beban. Perubahan beban ini akan membuat kecepatan aktual menjadi berbeda dengan kecepatan referensi.

Kemudian pada penelitian yang lain [2], mengenai *tracking* kecepatan pada HEV paralel dengan menggunakan *fuzzy controller*, pengujian dilakukan dengan *input* referensi kecepatan yang berubah-ubah. Hasil penelitian menunjukkan bahwa, *fuzzy controller* dapat membuat kecepatan aktual melakukan *tracking* kecepatan referensi. Tetapi jika terjadi kenaikan beban, maka akan menyebabkan penurunan kecepatan aktual. Kelemahan-kelemahan ini yang harus ditanggulangi. Oleh karena itu, diperlukan suatu strategi yang dapat mengatasi perubahan beban yang terjadi pada *parallel* HEV.

Pada suatu penelitian [3], perubahan beban ini bisa dideteksi menggunakan *disturbance observer*. Hasil menunjukkan bahwa *disturbance observer* dapat mengkompensasi perubahan beban, friksi, dan *modelling error* yang terjadi. Karena pada *disturbance observer* terdapat komponen *invers model*, maka diperlukan suatu metode untuk membuat *invers model* suatu *plant*. Pada penelitian lainnya [4], menyebutkan bahwa penggunaan *neuro-fuzzy* dapat digunakan untuk menggambarkan *invers model* suatu *plant* tanpa membuat model matematikanya. Dalam hasil penelitian tersebut menunjukkan *neuro-fuzzy invers model* memiliki *mean square error* yang lebih kecil dibandingkan *invers model* menggunakan *parametric model*. Oleh karena itu, agar kecepatan HEV bisa mengikuti kecepatan referensi dan dapat menolak gangguan, maka penulis memiliki ide untuk membuat *fuzzy controller* yang digabungkan dengan *disturbance observer* dimana *invers model*-nya menggunakan *neuro-fuzzy invers model*.

1.2 Rumusan Masalah

Dari penelitian-penelitian mengenai pengaturan kecepatan pada *parallel* HEV (PHEV) yang telah dijelaskan sebelumnya, menunjukkan bahwa kontroler *fuzzy* dapat membuat kecepatan aktual sama dengan kecepatan referensi selama tidak ada perubahan beban pada PHEV. Tetapi ketika PHEV mengalami perubahan beban, maka kontroler akan gagal dalam mempertahankan kecepatan aktual agar sama dengan kecepatan referensi. Kelemahan ini akan coba diatasi dengan menambahkan *disturbance observer* dimana memiliki kelebihan dalam mengkompensasi perubahan beban yang diterima suatu *plant*. Sedangkan komponen *invers model* pada *disturbance observer* akan menggunakan *neuro-*

fuzzy invers model dimana memiliki kelebihan dalam menggambarkan *invers* model sebuah *plant* tanpa membuat model matematikanya.

1.3 Tujuan

Seperti yang dijabarkan sebelumnya, maka dapat diambil tujuan utama dari penelitian ini yaitu memperbaiki kelemahan *fuzzy controller* dengan menambahkan *disturbance observer* dimana *invers model* pada *disturbance observer* ini menggunakan *neuro-fuzzy invers model* sehingga PHEV akan tahan terhadap perubahan beban (*disturbance*)

1.4 Batasan Masalah

Batasan masalah pada penelitian ini meliputi beberapa hal, yaitu sebagai berikut:

1. Penelitian ini menggunakan model *plant* PHEV yang diambil dari model yang tersedia pada *software* MATLAB 2014a.
2. Kendaraan berjalan lurus kedepan, tidak ada gerakan berbelok atau berjalan mundur.
3. Model *plant* PHEV akan diberi 1 gangguan (*disturbance*) yaitu gangguan yang diakibatkan oleh jalan dengan model dan kemiringan tertentu.

1.5 Kontribusi

Kontribusi yang ingin dicapai pada penelitian ini yaitu menerapkan metode baru *fuzzy disturbance observer* yang merupakan penggabungan antara kontroler *fuzzy* dan *disturbance observer* untuk pengaturan kecepatan pada *parallel* HEV dimana komponen *invers model disturbance observer* menggunakan *neuro-fuzzy invers model*.

1.6 Metodologi

Metodologi yang digunakan dalam penelitian ini antara lain:

1. Studi literatur

Studi literatur dilakukan dengan cara mengumpulkan dan mempelajari penelitian-penelitian dan buku-buku yang relevan terhadap topik yang akan diteliti. Penelitian-penelitian dan buku-buku ini berkaitan dengan *fuzzy controller*, *disturbance observer*, dan *neuro-fuzzy invers model*.

2. Perancangan sistem

Pada tahap ini akan dirancang suatu sistem kontrol *fuzzy* pada operasi normal. Setelah itu mencari bentuk *invers model plant parallel* HEV dengan *neuro-fuzzy invers model*. Kemudian *neuro-fuzzy invers model* ini dimasukkan ke dalam skema *disturbance observer*. Lalu *disturbance observer* ini akan digabungkan dengan kontroler *fuzzy* di mana hasilnya akan disimulasikan dengan perangkat lunak MATLAB 2014a.

3. Pengujian dan analisis sistem

Pada tahap pengujian dan analisis, sistem kontrol yang telah dirancang akan disimulasikan pada MATLAB 2014a dengan gangguan (*disturbance*) yang diakibatkan oleh jalan dengan model dan kemiringan tertentu. Hasil respon dari gangguan ini akan dicatat. Dari hasil pengujian ini dapat diperoleh data yang digunakan untuk menganalisis performa sistem kontrol yang telah dibuat.

4. Kesimpulan

Kesimpulan diperoleh sesuai dengan hasil pengujian dan analisis yang dilakukan.

5. Penulisan laporan tesis

Penulisan laporan tesis dilakukan sebagai dokumentasi dari hasil penelitian yang dilakukan.

BAB 2

KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1 Kajian Pustaka

Banyak penelitian bermunculan mengenai *hybrid electric vehicle* (HEV). Penelitian-penelitian tersebut banyak berfokus pada berbagai topik yaitu strategi pengaturan torsi, kecepatan, *power*, manajemen energi dan lain sebagainya. Penelitian ini akan berfokus pada salah satu topik yaitu strategi pengaturan kecepatan pada HEV bertipe paralel. Oleh karena itu, pembahasan kajian pustaka akan berfokus pada strategi pengaturan kecepatan pada *parallel* HEV.

Beberapa penelitian yang dikaji pada bab ini adalah penelitian mengenai penggunaan *fuzzy logic* yang mana digunakan sebagai kontrol dalam *parallel* HEV [1]. *Fuzzy controller* ini akan diterapkan untuk mengatur kecepatan *parallel* HEV. Dalam strategi kontrol yang diusulkan, ada dua variabel *fuzzy* input yaitu eror kecepatan dan perubahan eror kecepatan dan satu variabel *fuzzy* output yaitu sinyal kontrol kecepatan. Masing-masing variabel *fuzzy* tersebut memiliki tujuh variabel linguistik yaitu LN (*Large Negative*), MN (*Medium Negative*), SN (*Small Negative*), VS (*Very Small*), SP (*Small Positive*), MP (*Medium Positive*), dan LP (*Large Positive*) sehingga terdapat 49 aturan dalam *rule base*-nya. *Fuzzy controller* ini telah disimulasikan dengan dua kondisi, pertama, simulasi dilakukan dengan input kecepatan yang berubah dan torsi beban yang tetap, dan kedua, simulasi dilakukan dengan input kecepatan yang tetap dan torsi beban yang berubah. Pada simulasi yang pertama, kecepatan referensi berubah pada saat $t=3$ s dari kecepatan 1429 rpm menuju kecepatan 1214 rpm pada torsi beban yang tetap sebesar 10 N.m. Hasil simulasi menunjukkan bahwa kecepatan aktual hampir mengikuti kecepatan referensi, dimana kecepatan aktualnya sebesar 1143 rpm. Sedangkan pada simulasi kedua, kecepatan referensi tetap pada kecepatan 1200 rpm, dan kecepatan aktual mengikuti kecepatan referensi pada 1200 rpm juga. Kemudian torsi beban berubah pada saat $t = 3$ s yang semula bernilai 10 N.m menjadi 200 N.m. Hasil simulasi menunjukkan bahwa terjadi penurunan kecepatan sebesar 142 rpm sehingga menjadi 1058 rpm. Dari kedua simulasi ini

terlihat bahwa kecepatan aktual sistem mengikuti kecepatan referensi dengan sangat baik ketika torsi beban tidak berubah, tetapi pada saat torsi beban berubah pada kecepatan referensi yang sama, akan terjadi penurunan kecepatan aktual. Hal ini merupakan kelemahan yang akan diatasi pada penelitian ini.

Pada penelitian yang lain, pengujian dilakukan dengan menggunakan *fuzzy logic* untuk *tracking* kecepatan dari *parallel hybrid electric vehicle* [2]. *Fuzzy logic* ini mempunyai 2 variabel input dan 1 variabel output. Variabel input berupa eror kecepatan dan perubahan eror kecepatan sedangkan variabel output berupa sinyal kontrol yang akan mengatur arus yang masuk pada motor. Pada masing-masing variabel input dan variabel output ini terdapat 7 variabel linguistik yaitu NL (*Negative Large*), NM (*Negative Medium*), NS (*Negative Small*), ZO (*Zero*), PS (*Positive Small*), PM (*Positive Medium*), PL (*Positive Large*). Simulasi dilakukan sebanyak dua kali dengan kondisi yang berbeda. Pada simulasi yang pertama, simulasi ini dilakukan dengan dengan kecepatan yang berubah-ubah mulai dari kecepatan 2500 rpm menuju 3400 rpm lalu 1400 rpm kemudian 2800 rpm dengan torsi beban yang tetap yaitu 100 N.m. Hasilnya adalah *fuzzy controller* dapat melakukan *tracking* sesuai dengan kecepatan referensi yang berubah-ubah. Sedangkan pada simulasi kedua, simulasi dilakukan dengan membuat perubahan torsi beban yaitu dari 200 Nm menjadi 400 Nm pada kondisi kecepatan tetap yaitu 2300 rpm. Hasil menunjukkan bahwa terjadi penurunan kecepatan sebesar 18 rpm. Dari kedua hasil simulasi ini dapat disimpulkan bahwa *fuzzy controller* yang diterapkan pada motor dapat melakukan *tracking* kecepatan pada *parallel HEV* dengan syarat torsi beban yang tidak berubah.

Dalam mengatasi *disturbance* yang terjadi pada sebuah *plant*, maka terdapat satu penelitian mengenai penggunaan *disturbance observer* dalam mengatasi *disturbance* yang terjadi pada *plant* motor DC [3]. Pada pengujian ini, input referensi adalah arus yang mengalir ke DC motor sedangkan kontroler yang dipakai adalah kontroler PID. *Disturbance observer* pada desain ini digunakan untuk mengestimasi gangguan yang terjadi dimana gangguan ini akan menyebabkan arus yang mengalir pada motor tidak stabil. Desain ini diimplementasikan pada DC Motor tipe *brush* dengan dua kali percobaan. Pada percobaan pertama, nilai referensi dimasukkan sebesar 2A, karena terdapat torsi

beban friksi maka arus yang terukur tidak stabil. Dengan menggunakan *disturbance observer*, arus yang terukur menjadi stabil pada 2 ampere. Pada percobaan kedua, nilai referensi dimasukkan nilai yang berbeda, yaitu 0,5 A, dan gangguan eksternal ditambahkan selain torsi beban friksi. Arus yang terukur menuju 0,5 A tapi sangat tidak stabil, dengan ditambahkan *disturbance observer* maka arus yang terukur stabil sebesar 0,5 A. Dari kedua percobaan ini dapat disimpulkan bahwa, *disturbance observer* dapat menstabilkan output sistem ketika terjadi perubahan torsi beban.

Karena dalam pembuatan *disturbance observer* ini memerlukan *invers model* dari *plant* maka diperlukan suatu metode dalam pencarian *invers model* dari *plant* ini. Pada suatu penelitian [4], dibuat sebuah kontrol dengan menggunakan *neuro-fuzzy invers model* di mana *plant* yang digunakan adalah *plant* motor DC. Perancangan *neuro-fuzzy invers model* ini melalui tahapan training terlebih dahulu menggunakan 60000 data dan tes validasi menggunakan 40000 data. Kemudian setelah proses tersebut, kontrol menggunakan *neuro-fuzzy invers model* ini akan dibandingkan dengan menggunakan struktur identifikasi menggunakan *parametric model* secara *real time*. Hasilnya adalah penggunaan *neuro-fuzzy invers model* menghasilkan *Mean Square Error* (MSE) sebesar 1,19% sedangkan *parametric model* menghasilkan MSE sebesar 9,23%. Dari hasil ini dapat disimpulkan bahwa *neuro-fuzzy* lebih baik jika digunakan sebagai *invers model* daripada menggunakan *parametric model*.

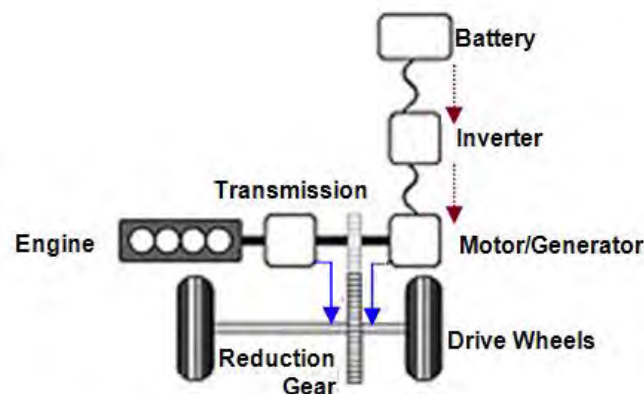
2.2 Landasan Teori

Pada landasan teori ini akan dibahas beberapa hal yang menyangkut tentang penelitian ini, yaitu *parallel hybrid electric vehicle*, *disturbance observer*, *fuzzy logic controller*, dan *neuro-fuzzy*.

2.2.1 Parallel Hybrid Electric Vehicle (HEV)

Hybrid Electric Vehicle atau yang disingkat HEV adalah kendaraan yang merupakan penggabungan dari mobil yang memakai *internal combustion engine* (ICE) dan kendaraan listrik (*electrical vehicle*) [5]. Karena memakai dua daya penggerak maka HEV membutuhkan dua sumber energi yang berasal dari bahan bakar fosil untuk ICE dan baterai untuk motor listrik. Ada beberapa cara untuk membuat konfigurasi antara daya penggerak motor listrik dan ICE. Dua konfigurasi utama pada HEV adalah HEV seri dan HEV paralel.

Dalam konfigurasi HEV paralel, pemindah daya (*drive train*) terhubung dengan motor listrik dan mesin melalui kopling mekanik. Kendaraan ini tidak memerlukan generator seperti dalam kasus konfigurasi HEV seri. Motor listrik dapat juga bertindak sebagai generator melalui kopling mekanik yang kemudian dapat digunakan untuk pengereman regeneratif. Konfigurasi HEV paralel ini diilustrasikan pada Gambar 2.1.



Gambar 2.1 Parallel HEV [6]

Konfigurasi HEV paralel ini memiliki kelebihan dan kekurangan, kelebihan konfigurasi ini yaitu sebagai berikut:

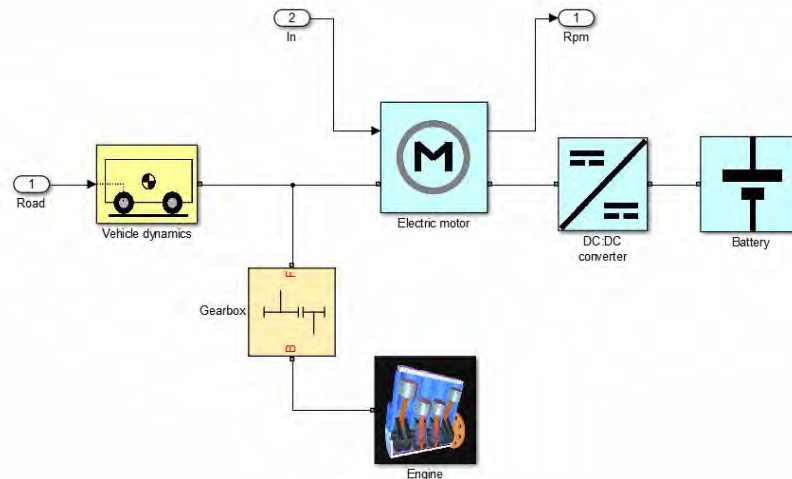
1. Dalam konfigurasi ini, baterai yang digunakan sebagai sumber daya listrik untuk motor listrik dapat berukuran kecil, karena kedua daya penggerak, mesin dan motor listrik terhubung pada pemindah daya (*drive train*).

2. Kemampuan HEV paralel ini sangat baik jika dibandingkan dengan kendaraan konvensional karena HEV paralel menggunakan sumber penggerak ganda.
3. Kendala keterbatasan baterai dan motor listrik dalam konfigurasi ini tidak perlu menjadi perhatian utama.

Sedangkan kekurangan konfigurasi ini adalah sebagai berikut:

1. Kendaraan tidak bisa mendapatkan daya akselerasi penuh dari motor listrik ketika daya listrik baterai hampir habis.
2. Kompleksitas kontrol meningkat secara signifikan, karena aliran daya harus diatur dari dua sumber yang paralel.
3. Daya penggerak dari ICE dan motor memerlukan perangkat mekanis kompleks.

Pada penelitian ini kontroler yang dibangun akan diaplikasikan pada sebuah model *plant parallel* HEV. Model *plant* ini didapat dari simulink pada MATLAB R2014a. Diagram *simulink* model *plant* tersebut diperlihatkan pada Gambar 2.2 [7].



Gambar 2.2 Diagram *Simulink Plant Parallel HEV*

Pada simulasi di atas, terdapat 6 blok bagian yang selanjutnya akan dijelaskan sebagai berikut [7]:

1. *Electric Motor*

Electric motor merupakan penggerak sekunder dari *parallel* HEV. Pada perjalanan pendek, *Electric motor* ini akan menjadi penggerak utama dalam *parallel* HEV. Selain pada perjalanan pendek, *electric motor* juga digunakan untuk membantu *engine* ketika harus melewati tanjakan atau saat akselerasi di jalan raya. Selain digunakan sebagai sumber penggerak, bisa juga digunakan sebagai generator untuk *charging* baterai dimana proses *charging* ini akan dilakukan saat pengereman regeneratif.

Pada simulasi model *plant parallel* HEV ini, *electric motor* tidak digunakan sebagai generator sehingga pengereman regeneratif tidak disimulasikan. Kecepatan sudut motor ini akan diukur menggunakan sensor kecepatan angular. Daya motor dan daya motor yang hilang selama *electric motor* bekerja, dapat dihitung menggunakan persamaan daya yang ditunjukkan dengan (2.1) dan (2.2)

$$P_{\text{motor}} = T \cdot \omega \quad (2.1)$$

$$P_{\text{loss}} = k \cdot T^2 \quad (2.2)$$

dengan:

P_{motor} = daya motor (W)

T = torsi motor (Nm)

ω = kecepatan sudut motor (rad/s)

k = konstanta ($\text{W/N}^2\text{m}^2$)

P_{loss} = daya motor yang hilang (W)

Saat motor bekerja, arus yang mengalir pada motor dihitung dengan persamaan yang ditunjukkan pada (2.3)

$$I = \frac{P_{\text{motor}} + \text{sign}(P_{\text{motor}}) \cdot P_{\text{loss}}}{V} \quad (2.3)$$

dengan:

I = arus yang mengalir pada motor (A)

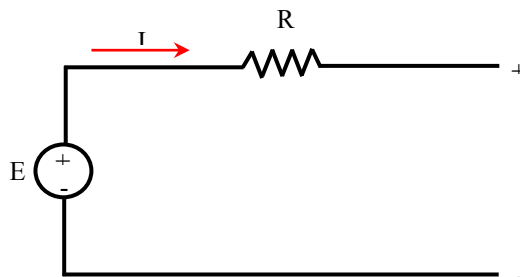
V = tegangan motor (V)

Arus ini akan dibutuhkan untuk memperhitungkan daya yang digunakan oleh baterai dan daya baterai yang hilang. Sedangkan parameter motor yang digunakan pada simulasi ini adalah sebagai berikut:

$$V = 500 \text{ V}$$

2. Battery

Baterai merupakan salah satu sumber energi listrik. Dalam HEV, baterai ini merupakan komponen yang penting dimana akan dikonversi ke energi mekanik melalui motor listrik. Fungsi lain baterai pada HEV adalah sebagai tempat menyimpan energi listrik yang dihasilkan dari generator atau pengereman regeneratif. Terdapat beberapa model dalam menggambarkan karakteristik baterai. Pada model *parallel* HEV ini, model baterai yang digunakan adalah *linier model*. Pada *linier model* ini resistansi internal baterai akan diperhitungkan. Gambar 2.3 menunjukkan karakteristik baterai dengan menggunakan *linier model*.



Gambar 2.3 *Linier Model*

Pada Gambar 2.3, E adalah tegangan baterai dan R adalah hambatan internal baterai.

Pada baterai ini akan dilihat daya baterai dan daya baterai yang hilang dimana persamaan yang menggambarkan keduanya ditunjukkan pada persamaan (2.4) dan (2.5)

$$P_{\text{battery}} = V \cdot I \quad (2.4)$$

$$P_{\text{loss}} = I^2 R \quad (2.5)$$

dengan:

P_{batt} = daya baterai (W)

V = tegangan baterai (V)

I = arus yang mengalir (A)

R = hambatan internal baterai (Ω)

P_{loss} = daya baterai yang hilang (W)

Pada simulasi ini, *battery* akan dimodelkan dengan parameter sebagai berikut:

$$V = 201,6 \text{ V}$$

$$R = 0,05 \Omega$$

3. DC-DC Converter

DC-DC *converter* merupakan sebuah rangkaian yang mengubah tegangan *input* DC yang tidak teregulasi pada level tegangan tertentu, ke tegangan *output* DC yang teregulasi pada level tegangan yang berbeda dengan efisiensi yang tinggi.

Pada simulasi model plant parallel HEV ini menggunakan blok DC-DC converter. DC-DC converter ini memiliki fungsi utama yaitu menaikkan tegangan battery. Tegangan battery adalah 201,6 V, sedangkan electric motor membutuhkan tegangan sebesar 500 V. Oleh karena itu blk DC-DC converter ini mutlak diperlukan.

Pada blok DC-DC converter ini menggunakan pendekatan ideal transformer model. Ideal transformer model ini memiliki 2 lilitan yaitu lilitan primer dan lilitan sekunder. Karena DC-DC converter didekati dengan ideal transformer model, maka persamaan-persamaan transformator berlaku pada blok DC-DC converter ini. Hubungan antara tegangan primer dan sekunder serta hubungan antara arus primer dan sekunder digambarkan dengan (2.6) dan (2.7).

$$V_1 = N \cdot V_2 \quad (2.6)$$

$$I_2 = N \cdot I_1 \quad (2.7)$$

dengan:

V_1 = tegangan primer (V)

V_2 = tegangan sekunder (V)

I_1 = arus primer (V)

I_2 = arus sekunder (V)

N = ratio belitan

Jadi pada simulasi ini DC-DC *Converter* akan disimulasikan dengan menggunakan parameter sebagai berikut:

$$N = 2,48$$

4. *Gearbox*

Pada *parallel* HEV, tenaga yang berasal dari dua penggerak mesin yaitu *engine* dan motor listrik harus disatukan. Hal ini dikarenakan poros penanggung beban utama yang merupakan poros *engine*, mempunyai letak yang terpisah dengan poros penggerak sekunder yang merupakan poros motor DC. Untuk menyatukan kedua poros penggerak yang berlainan ini maka dibutuhkan *gearbox*. Pada simulasi model *plant parallel* HEV ini, *gearbox* akan disimulasikan dengan sebuah *simple gear* dimana mempunyai dua poros yang dihubungkan yaitu *base* (B) dan

follower (F). Poros *base* akan dihubungkan ke *engine* sedangkan poros *follower* akan dihubungkan ke motor listrik.

Pada *simple gear*, persamaan kinematik yang menghubungkan dua poros yang berbeda ditunjukkan pada (2.8) dan (2.9).

$$r_F \omega_F = r_B \omega_B \quad (2.8)$$

$$g_{FB} = \frac{r_F}{r_B} = \frac{N_F}{N_B} \quad (2.9)$$

dengan:

- r_F = jari-jari *gear follower* (m)
- ω_F = kecepatan sudut *gear follower* (rad/s)
- r_B = jari-jari *gear follower* (m)
- ω_B = kecepatan sudut *gear follower* (rad/s)
- N_F = jumlah gigi pada *gear follower*
- N_B = jumlah gigi pada *gear base*
- g_{FB} = *follower-base gear ratio*

Sedangkan persamaan torsi yang menghubungkan *gear base* dan *gear follower* ditunjukkan pada (2.10).

$$g_{FB} T_B + T_F - T_{\text{loss}} = 0 \quad (2.10)$$

Untuk kondisi *ideal gear* maka $T_{\text{loss}} = 0$ sedangkan untuk kondisi *nonideal gear* maka $T_{\text{loss}} \neq 0$. Persamaan yang mendefinisikan T_{loss} ini ditunjukkan pada (2.11) dan (2.12).

$$T_{\text{loss}} = T_{\text{coul}} \cdot \tanh \left(4 \frac{\omega_{\text{out}}}{\omega_{\text{th}}} \right) + \mu \omega_{\text{out}} \quad (2.11)$$

$$T_{\text{coul}} = |T_F| (1 - \eta) \quad (2.12)$$

dengan:

- T_{loss} = torsi yang hilang (Nm)

T_{coul}	= torsi coulomb (Nm)
ω_{out}	= kecepatan sudut <i>output</i> (rad/s)
ω_{th}	= kecepatan sudut <i>threshold gear follower</i> (rad/s)
μ	= koefisien <i>viscous friction</i> (Nm.s/rad)
T_F	= torsi <i>gear follower</i> (Nm)
η	= efisiensi

Pada model *plant parallel* HEV ini, *gearbox* memiliki beberapa parameter yaitu sebagai berikut:

g_{FB}	= 4,27
η	= 0,95
ω_{th}	= 0,01 rad/s
μ	= 0 Nm.s/rad

5. *Engine*

Engine merupakan penggerak utama dalam *parallel* HEV. Pada simulasi ini, *engine* dimodelkan dengan menggunakan *engine power demand function* yaitu $g(\Omega)$. Fungsi tersebut menghasilkan nilai daya maksimum untuk kecepatan *engine* sebesar Ω . Terdapat *input throttle* yang ternormalisasi yang akan menentukan daya mesin aktual. Daya akan dihitung sebagai pecahan dari daya maksimum yang mungkin pada kondisi stabil di kecepatan *engine* yang tetap. Daya maksimum *engine* P_{max} akan dihitung berdasarkan nilai Ω_0 sehingga aturan $P_{\text{max}} = g(\Omega_0)$ terpenuhi. Dengan mendefinisikan $w = \Omega / \Omega_0$ dan $g(\Omega) = P_{\text{max}} p(w)$ maka $p(1) = 1$ dan $dp(1)/dw = 0$. Sedangkan fungsi torsi digambarkan dengan (2.13)

$$\tau = \left(\frac{P_{\text{max}}}{\Omega_0} \right) \left[\frac{p(w)}{w} \right] \quad (2.13)$$

Pada model simulasi *engine* ini akan digunakan bentuk polinomial orde tiga yang ditunjukkan pada (2.14)

$$p(w) = p_1w + p_2w^2 - p_3w^3 \quad (2.14)$$

Dimana pemilihan p_1 , p_2 , dan p_3 harus memenuhi persamaan yang ditunjukkan pada (2.15) dan (2.16)

$$p_1 + p_2 - p_3 = 1 \quad (2.15)$$

$$p_1 + 2p_2 - 3p_3 = 0 \quad (2.16)$$

Pada beberapa *engine*, semua parameter p adalah positif. Polinomial orde 3 ini memiliki 3 penyelesaian, yaitu $w=0$ dan sepasang *conjugate* yang memiliki nilai yang ditunjukkan pada (2.17).

$$w_{\pm} = \frac{1}{2}(-p_2 \pm \sqrt{p_2^2 + 4p_1p_3}) \quad (2.17)$$

Pada simulasi ini, terdapat 2 pilihan jenis *engine* yaitu *spark ignition* dan *diesel*. Parameter-parameter pada kedua jenis *engine* ini dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Koefisien pada 2 Jenis *Engine*

Koefisien	<i>Spark-Ignition</i>	<i>Diesel</i>
p_1	1	0,6526
p_2	1	1,6948
p_3	1	1,3474

Engine pada simulasi ini memiliki parameter sebagai berikut:

$$P_{\max} = 50000 \text{ W}$$

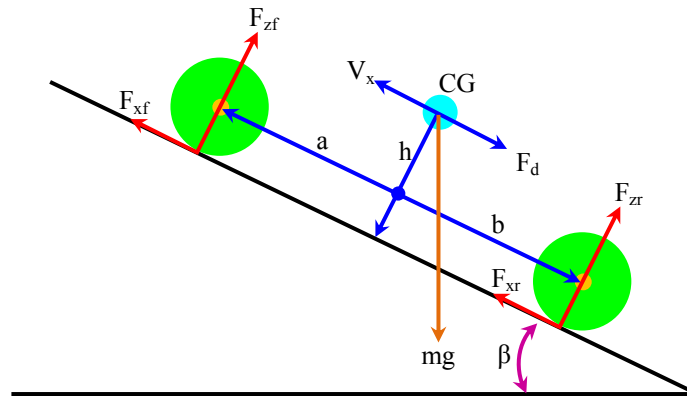
$$\omega = 5500 \text{ rpm}$$

$$\omega_{\max} = 7000 \text{ rpm}$$

$$\omega_{\text{stall}} = 500 \text{ rpm}$$

6. *Vehicle Dynamic*

Salah satu aspek penting dalam simulasi model *plant parallel* HEV ini adalah membuat simulasi kendaraan. Hal ini penting karena dua penggerak yang berbeda bekerja secara paralel untuk menggerakkan kendaraan melaju dengan kecepatan yang diinginkan. Pada simulasi ini kendaraan dimodelkan seperti pada Gambar 2.4.



Gambar 2.4 Model Kendaraan

dengan:

- m = massa kendaraan (kg)
- g = percepatan gravitasi (m/s^2)
- β = sudut tanjakan ($^\circ$)
- h = jarak antara titik berat kendaraan terhadap tanah (m)
- V_x = kecepatan longitudinal kendaraan (m/s)
- a = jarak as roda kendaraan depan ke titik potong antara garis h dengan garis jarak antara as roda depan dan belakang (m)
- b = jarak as roda kendaraan belakang ke titik potong antara garis h dengan garis jarak antara as roda depan dan belakang (m)
- F_{xf} = Gaya longitudinal roda depan terhadap titik kontak roda depan terhadap tanah (N)
- F_{xr} = Gaya longitudinal roda belakang terhadap titik kontak roda belakang terhadap tanah (N)

- F_{zf} = Gaya normal roda depan terhadap titik kontak roda depan terhadap tanah (N)
 F_{zr} = Gaya normal roda belakang terhadap titik kontak roda belakang terhadap tanah (N)
 F_d = gaya hambat aerodinamik (N)

Pada bentuk model di atas, poros-poros roda akan sejajar dan membentuk sebuah bidang. Arah sumbu x akan berada pada bidang ini dan tegak lurus arah poros roda. Jika kendaraan berjalan di atas jalan dengan kemiringan dengan sudut β maka arah gaya berat tidak lagi searah dengan sumbu z tetapi tegak lurus dengan bidang datar. Persamaan yang menggambarkan dinamika pergerakan kendaraan bisa disederhanakan seperti yang ditunjukkan pada (2.18), (2.19), dan (2.20).

$$m\dot{V}_x = F_x - F_d - mg \sin \beta \quad (2.18)$$

$$F_x = n(F_{xf} + F_{xr}) \quad (2.19)$$

$$F_d = \frac{1}{2} C_d \rho A (V_x - V_w)^2 \cdot \text{sign}(V_x - V_w) \quad (2.20)$$

Sedangkan gaya normal yang terjadi pada roda depan dan roda belakang kendaraan, ditunjukkan ada (2.21) dan (2.22)

$$F_{zf} = \frac{-h(F_d + mg \sin \beta + m\dot{V}_x) + b \cdot mg \cos \beta}{n(a+b)} \quad (2.21)$$

$$F_{zr} = \frac{+h(F_d + mg \sin \beta + m\dot{V}_x) + a \cdot mg \cos \beta}{n(a+b)} \quad (2.22)$$

Dengan persamaan gaya normal tiap roda ditunjukkan pada (2.23)

$$F_{zf} + F_{zr} = \frac{mg \cos \beta}{n} \quad (2.23)$$

dengan:

A = luas efektif kendaraan (m^2)

C_d = koefisien gaya hambat aerodinamik ($kg \cdot m/s^4$)

ρ = massa jenis udara (kg/m^3)

Pada simulasi ini, kendaraan memiliki parameter sebagai berikut:

m = 1200 kg

h = 0,8 m

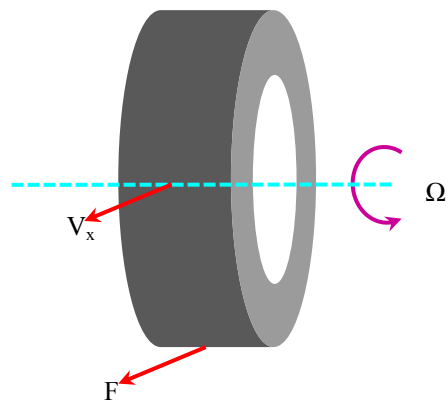
a = 1,4 m

b = 1,6 m

A = 3 m^2

C_d = 0,4 $kg \cdot m/s^4$

Selain dinamika kendaraan, roda kendaraan juga disimulasikan pada model *plant parallel* HEV ini. Pada simulasi ini, roda kendaraan akan dimodelkan seperti pada Gambar 2.5.



Gambar 2.5 Model Roda Kendaraan

Jika roda tidak mengalami slip, maka roda tersebut akan bergulir dan bergerak translasi yang besarnya sesuai dengan (2.24).

$$V_x = r_w \Omega \quad (2.24)$$

dengan:

V_x = kecepatan longitudinal roda (m/s)

r_w = radius roda (m)

Ω = kecepatan sudut roda (rad/s)

Pada keadaan sebenarnya, roda akan mengalami slip sehingga timbul gaya longitudinal F_x sebagai akibat dari slip yang terjadi. Ketika mengalami slip, roda akan melakukan gerak translasi dengan kecepatan yang sesuai dengan (2.25).

$$V_{sx} = r_w \Omega - V_x \quad (2.25)$$

dengan:

V_{sx} = kecepatan longitudinal roda ketika slip (m/s)

Dengan konstanta slip yang memenuhi aturan (2.26).

$$\kappa = \frac{V_{sx}}{|V_x|} \quad (2.26)$$

dengan:

κ = konstanta slip

Ketika roda terkunci maka konstanta slip sebesar -1 sedangkan ketika berputar sempurna adalah sebesar 0. Tetapi jika ketika roda bergerak dengan kecepatan rendah, yaitu ketika $|V_x| \leq |V_{th}|$ maka konstanta slip roda memenuhi aturan (2.27).

$$\kappa = \frac{2V_{sx}}{V_{th} + \frac{V_x^2}{V_{th}}} \quad (2.27)$$

dengan:

$$V_{th} = \text{Kecepatan ambang (m/s)}$$

Pada simulasi ini, roda kendaraan diasumsikan hanya bergerak secara longitudinal dan tidak berbelok. Roda kendaraan ini memiliki parameter sebagai berikut:

$$\begin{aligned} r_w &= 0,3 \text{ m} \\ I &= 0,001 \text{ kg.m}^2 \\ V_{th} &= 0,1 \text{ m/s} \\ F_{ver} &= 3000 \text{ N} \\ F_{hor} &= 3500 \text{ N} \\ \kappa &= 0,1 \end{aligned}$$

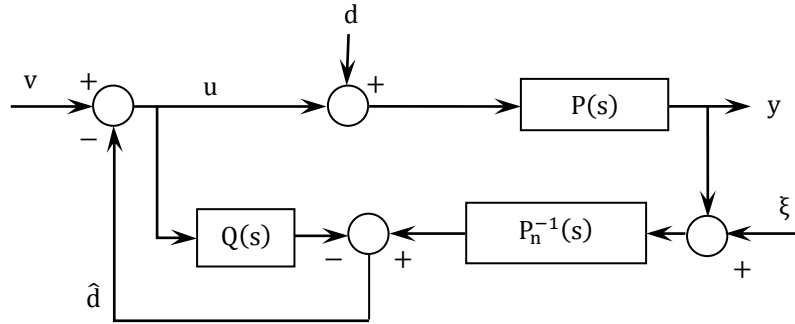
2.2.2 Disturbance Observer

Dalam desain kontrol gerak, *disturbance* merupakan permasalahan yang sangat penting selain *tracking performance*. *Disturbance observer* merupakan metode efektif yang sering digunakan untuk menangani masalah ini [8]. Pada desain *disturbance observer*, *inner loop* dari *disturbance observer* ditambahkan ke dalam *loop feedback* pada kontrol.

Gambar 2.6 menunjukkan struktur umum dari *disturbance observer* untuk *plant* SISO, di mana v , u , d , y dan ξ adalah *input* perintah, *input* kontrol, *disturbance* eksternal, *output* dan *sensor noise*. $P(s)$ merupakan *plant* fisik yang akan dikendalikan. $P_n(s)$ adalah model *plant* nominal. Dalam prakteknya, $P_n(s)$ dipilih sebagai pendekatan orde terendah *plant* fisik.

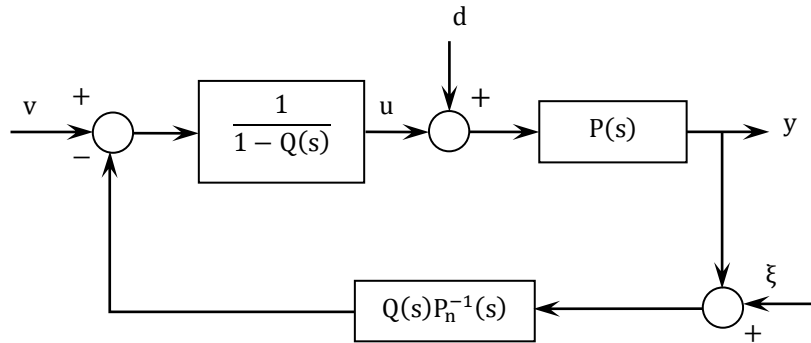
Disturbance observer menganggap ketidaksesuaian antara *plant* dan model nominal sebagai sebuah gangguan yang bekerja pada model nominal.

Disturbance observer memperkirakan gangguan yang setara dengan gangguan eksternal, dan mengembalikan kembali estimasi *disturbance* sebagai *cancellation signal*, seperti ditunjukkan pada Gambar 2.6.



Gambar 2.6 Struktur *Disturbance Observer*

Struktur *disturbance observer* pada Gambar 2.6, dapat disederhanakan menjadi Gambar 2.7.



Gambar 2.7 Struktur Penyederhanaan *Disturbance Observer*

Perilaku *loop* gangguan dapat dianalisis dengan melihat pada fungsi transfer dari v , d , dan ξ ke *output* y seperti yang ditunjukkan pada (2.28).

$$y = G_{yv}(s)v + G_{yd}(s)d + G_{y\xi}(s)\xi \quad (2.28)$$

Dari penyederhanaan struktur *disturbance observer* pada Gambar 2.7 dapat diturunkan menjadi (2.29), (2.30) dan (2.31).

$$G_{yv} = \frac{Y(s)}{v(s)} = \frac{P(s)P_n(s)}{P_n(s)+Q(s)[P(s)-P_n(s)]} \quad (2.29)$$

$$G_{yd} = \frac{Y(s)}{D(s)} = \frac{P(s)P_n(s)[1-Q(s)]}{P_n(s)+Q(s)[P(s)-P_n(s)]} \quad (2.30)$$

$$G_{y\xi} = \frac{Y(s)}{\xi(s)} = \frac{P(s)Q(s)}{P_n(s)+Q(s)[P(s)-P_n(s)]} \quad (2.31)$$

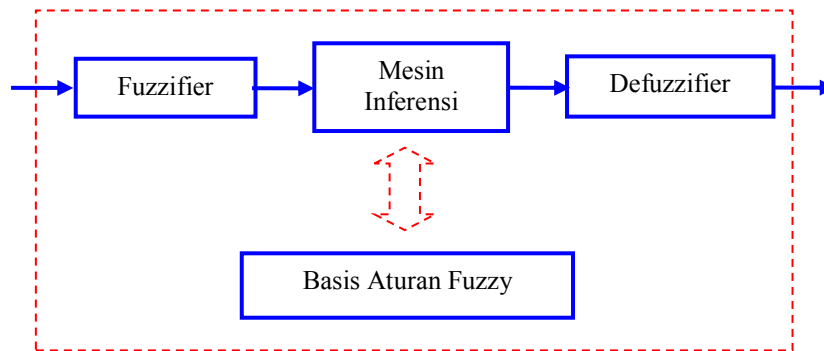
Ketika $Q(s) \approx 1$, persamaan 2.28 menjadi $y \approx P_n(s)v + \xi$. Hal ini menunjukkan bahwa *disturbance observer* menolak gangguan dan mengkompensasi ketidakcocokan model. Sebenarnya cara kerja *disturbance observer* berperilaku sebagai model nominal. Di sisi lain, ketika $Q(s) \approx 0$, persamaan 2.24 menjadi $y \approx P_n(s)v + P_n(s)d$. Hal ini mengakibatkan *loop disturbance observer* seakan-akan tidak ada, dan *output* tidak terpengaruh oleh *noise sensor*.

Dalam aplikasi kontrol, gangguan biasanya didominasi pada frekuensi rendah, sedangkan *noise sensor* didominasi pada frekuensi tinggi. Hal ini menunjukkan bahwa $Q(s)$ seharusnya merupakan *low pass filter*. Selain karakteristik *low pass filter* dari $Q(s)$, untuk *disturbance observer* pada Gambar 2.8 dapat direalisasi dengan catatan bahwa orde dari $Q(s)$ harus lebih besar dari atau sama dengan $P_n(s)$.

2.2.3 Fuzzy Logic Controller

Fuzzy Logic Controller (FLC) merupakan sistem *fuzzy* yang diaplikasikan secara khusus pada bidang kontrol. FLC banyak diaplikasikan pada bidang kontrol karena sifatnya yang lebih humanis dibandingkan dengan sistem kontrol klasik karena FLC merepresentasikan pengetahuan operator dalam mengoperasikan atau mengendalikan *plant* yang akan dikendalikan [9].

Variabel *input* pada *fuzzy controller* umumnya berupa nilai selisih antara nilai referensi *output* dengan nilai *output* aktual yang disebut nilai eror. Sedangkan *output fuzzy controller* adalah perintah kendali yang diberikan ke aktuator atau penggerak. Struktur *fuzzy controller* sama dengan sistem *fuzzy* pada umumnya seperti yang digambarkan pada Gambar 2.8.



Gambar 2.8 Struktur *Fuzzy Controller*

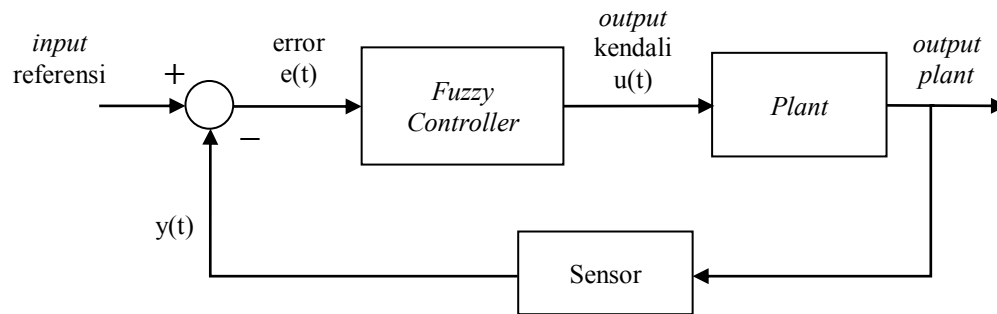
Bagian pertama disebut *fuzzifier* yang akan melakukan fuzzifikasi terhadap nilai variabel *input*. Basis aturan *fuzzy* berisi aturan-aturan yang menggambarkan bagaimana proses kendali dilakukan. Bagian mesin inferensi berfungsi mengambil kesimpulan terhadap nilai *input* yang ada berdasarkan aturan-aturan pada *rule base*, bagian akhir berupa *defuzzier* berfungsi melakukan defuzzifikasi untuk mengubah hasil inferensi *fuzzy* menjadi *output* yang siap dikirim ke aktuator. Berdasarkan tipe konsekuen yang digunakan pada *fuzzy rule base*, tipe *fuzzy controller* dapat dibedakan menjadi dua yaitu tipe mamdani dan tipe sugeno.

Fuzzy controller tipe mamdani lebih tepat digunakan jika model *plant* tidak diketahui secara pasti, tetapi terdapat operator atau pakar yang mengetahui perilaku dan karakteristik sistem yang dikendalikan. Sedangkan tipe sugeno lebih cocok digunakan pada kondisi tidak terdapat operator atau pakar, tetapi model matematis *plant* diketahui. *Fuzzy controller* dapat diterapkan pada sistem *open loop* maupun *closed loop*.

Pada struktur *closed loop* yang ditunjukkan pada Gambar 2.9, harus dipilih variabel *input*, variabel keadaan, maupun *output* sistem yang akan dikendalikan sebagai bagian dari anteseden aturan maupun konsekuen aturan. Variabel keadaan

yang bisa dipilih sebagai anteseden aturan *fuzzy* adalah *error* (e), *change of error* (Δe), dan *sum of error* (Σe).

Sedangkan variabel *input* sistem yang merupakan *output* pengendali dan merupakan bagian konsekuen aturan *fuzzy* dapat berupa *change of output control* (Δu) dan *output control* (u).



Gambar 2.9 Struktur *Fuzzy Controller* pada Sistem *Closed Loop*

2.2.4 Neuro-Fuzzy

Sistem *neuro-fuzzy* merupakan salah satu struktur pemodelan yang menggabungkan sistem *fuzzy* dan sistem jaringan syaraf tiruan [10]. Dasar dari penggabungan adalah kelebihan dan kekurangan dari masing-masing sistem. Kemampuan utama jaringan syaraf tiruan adalah dapat mengenali sistem melalui proses pembelajaran untuk memperbaiki parameter adaptif. Kekurangan dari sistem ini adalah kerumitan strukturnya.

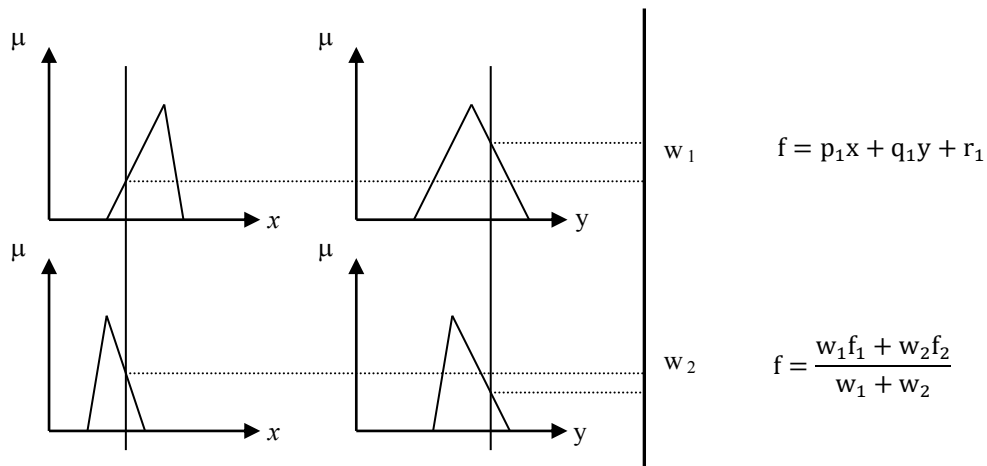
Sedangkan sistem *fuzzy* mempunyai konsep yang mirip dengan konsep berpikirnya manusia. Gabungan keduanya akan saling melengkapi kelebihan dan kekurangan masing-masing sistem. Sistem inferensi *fuzzy* yang digunakan adalah sistem *fuzzy* model TSK orde satu dengan pertimbangan kesederhanaan serta kemudahan komputasi.

Pada sistem inferensi *fuzzy* TSK orde satu dengan dua masukan, aturan yang digunakan adalah:

Aturan 1: **If** x is A_1 and y is B_1 **then** $f_1 = p_1x + q_1y + r_1$

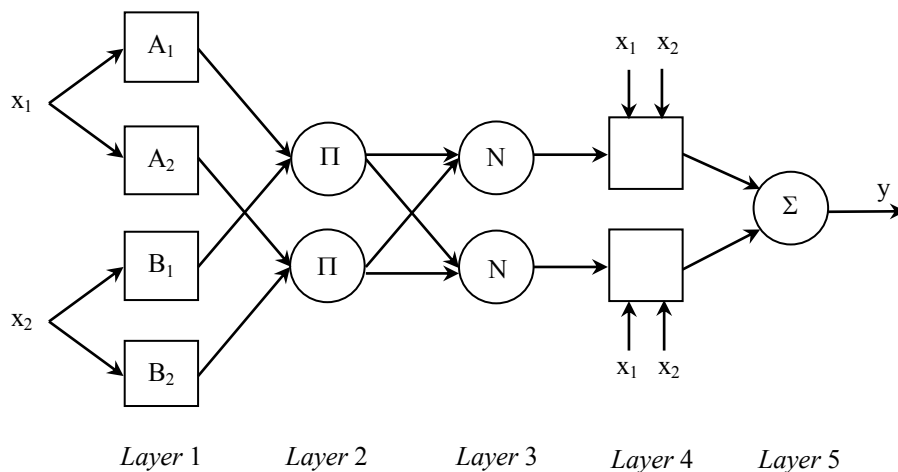
Aturan 2: **If** x is A_2 and y is B_2 **then** $f_2 = p_2x + q_2y + r_2$

Mekanisme sistem inferensi ini dapat dilihat pada Gambar 2.10.



Gambar 2.10 Sistem Inferensi *Fuzzy* TSK

Sistem *neuro-fuzzy* yang ekivalen dengan sistem inferensi *fuzzy* di atas mempunyai struktur jaringan dengan lima lapisan (*layer*) seperti yang ditunjukkan pada Gambar 2.11.



Gambar 2.11 Struktur *Neuro-Fuzzy*

Tiap *layer* memiliki mekanisme yang berbeda-beda mekanisme dijelaskan sebagai berikut

1. *Layer*

Layer 1 berfungsi sebagai fuzzyfikasi. *Output* pada *node* i pada *layer* 1 dinotasikan sebagai $O_{1,i}$.

Setiap *node* pada *layer* 1 bersifat adaptif dengan *output* ditunjukkan pada (2.32) dan (2.33).

$$O_{1,i} = \mu_{Ai}(x_1), i = 1,2 \quad (2.32)$$

$$O_{1,i} = \mu_{Bi}(x_2), i = 3,4 \quad (2.33)$$

Dimana x dan y adalah nilai-nilai *input* untuk *node* tersebut dan A_i atau B_i adalah himpunan *fuzzy*. Jadi masing-masing *node* pada *layer* 1 berfungsi membangkitkan derajat keanggotaan.

2. *Layer* 2

Layer 2 dinotasikan π . Setiap *node* pada *layer* ini berfungsi untuk menghitung kekuatan aktivasi (*firing strength*) pada setiap *rule* sebagai *product* dari semua *input* yang masuk. *Output* dari *layer* ini ditunjukkan pada (2.34), (2.35), dan (2.36).

$$O_{2,i} = w_i = \mu_{Ai}(x_1)\mu_{Bi}(x_2), i = 1,2 \quad (2.34)$$

$$w_1 = \mu_{A1}(x_1)AND\mu_{B1}(x_2) \quad (2.35)$$

$$w_2 = \mu_{A2}(x_1)AND\mu_{B2}(x_2) \quad (2.36)$$

3. *Layer* 3

Layer 3 dilambangkan dengan N . Setiap *node* pada lapisan ini bersifat non-adaptif yang berfungsi hanya untuk menghitung rasio antara *firing strength* pada *rule* ke- i terhadap total *firing strength* dari semua *rule*. *Output* dari *layer* 3 ini ditunjukkan pada (2.37).

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1+w_2}, i = 1,2 \quad (2.37)$$

4. *Layer 4*

Pada layer ini setiap *node* bersifat adaptif sebagai fungsi dari (2.38).

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x_1 + q_i x_2 + r_i) \quad (2.38)$$

Dimana \bar{w}_i adalah *output* dari *layer 3* dan $\{p_i x_1 + q_i x_2 + r_i\}$ adalah himpunan parameter pada *fuzzy model Sugeno orde pertama*.

5. *Layer 5*

Satu *node* tunggal yang dilambangkan dengan Σ . Pada *layer* ini berfungsi mengagregasikan seluruh *output* dari *layer 4* yang ditunjukkan pada (2.39).

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (2.39)$$

Kemudian dalam proses *learning* ANFIS diterapkan metode pembelajaran hibrid. Pembelajaran ini terdiri atas dua bagian, yaitu:

1. Arah maju (*forward*). Di sini parameter premis dibuat tetap. Dengan menggunakan metode *Least Square Estimator* (LSE), parameter konsekuen diperbaiki berdasarkan pasangan data pada *training set*. Metode LSE dapat diterapkan, karena parameter konsekuen yang diperbaiki bersifat linier. Setelah parameter konsekuen diperoleh, data masukan dilewatkan jaringan adaptif kembali dan hasil keluaran jaringan adaptif ini dibandingkan dengan keluaran yang diharapkan (*target*).
2. Arah mundur (*backward*). Di sini parameter konsekuen dibuat tetap. Kesalahan yang terjadi antara keluaran jaringan adaptif dan keluaran sebenarnya dipropagasikan balik dengan menggunakan metoda *gradient descent* untuk memperbarui parameter premis. Satu tahap pembelajaran maju-mundur ini dinamakan 1 *epoch*.

BAB III

PERANCANGAN SISTEM

3.1 Gambaran Umum Perancangan Sistem

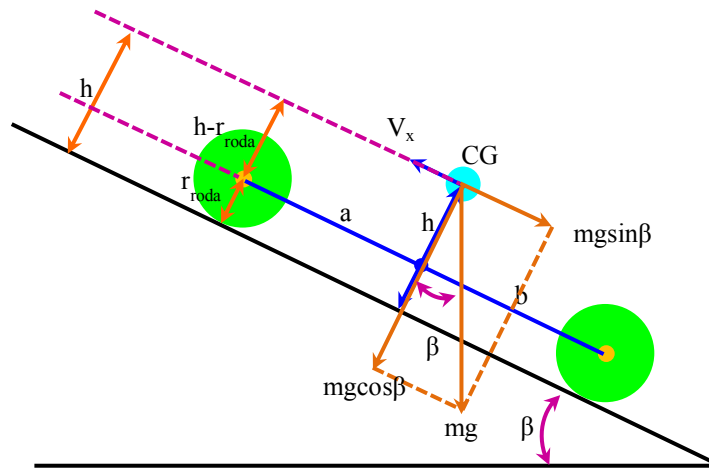
Pada bab ini akan dijelaskan mengenai 2 hal yaitu perancangan *disturbance*, dan perancangan kontrol. Dimana perancangan kontrol pada tesis ini yaitu *disturbance observer* dengan *neuro-fuzzy inverse model* terdiri dari beberapa tahapan yaitu perancangan *fuzzy controller*, *disturbance observer*, dan *neuro-fuzzy invers model*. Semua perancangan ini akan menggunakan simulasi model *Parallel Hybrid Electric Vehicle* (PHEV) seperti yang dijelaskan pada bab sebelumnya.

3.1.1 Perancangan *Disturbance*

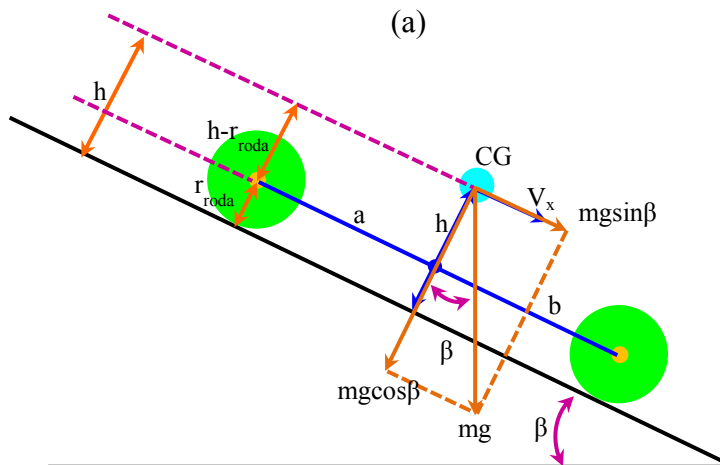
Pada penelitian ini, PHEV akan berada pada lingkungan dengan *disturbance* yang tinggi dengan referensi kecepatan yang tetap. Oleh karena itu, tujuan utama penelitian ini adalah menerapkan kontroler *fuzzy* dan *disturbance observer* dengan *neuro-fuzzy invers model* agar *parallel HEV* memiliki kecepatan yang tetap walaupun kendaraan menerima *disturbance* yang datang dari lingkungan. Oleh karena itu, dalam simulasi ini diperlukan *disturbance* agar dapat menguji kemampuan kontroler ini terhadap *disturbance* yang datang.

Pada penelitian ini, digunakan *disturbance* (gangguan) yang diakibatkan oleh jalan yang dilalui model PHEV dimana jalan tersebut memiliki kemiringan tertentu. *Disturbance* yang diakibatkan oleh kemiringan jalan diambil karena hal tersebut berpengaruh kepada kecepatan kendaraan. Jalan yang menurun akan cenderung membuat kendaraan bertambah cepat. Hal ini dikarenakan proyeksi gaya berat kendaraan akan mendorong kendaraan ke bawah sehingga kecepatan kendaraan akan naik. Dalam menanggulangi hal ini, putaran mesin *parallel HEV* harus diturunkan agar kecepatan kendaraan menjadi tetap. Begitu pula sebaliknya, jalan yang naik akan cenderung membuat kendaraan bertambah lambat. Hal ini dikarenakan proyeksi gaya berat kendaraan akan menarik kendaraan ke bawah sehingga kecepatan kendaraan akan lambat.

Penjelasan tersebut dapat digambarkan pada Gambar 3.1.



(a)



(b)

Gambar 3.1 Komponen Gaya Kendaraan Saat Melaju
(a) di Tanjakan (b) di Turunan

Gambar 3.1 menunjukkan bahwa komponen gaya berat $mg \sin \beta$ yang mempengaruhi kecepatan kendaraan. Komponen gaya berat $mg \sin \beta$ akan memberikan torsi gangguan pada roda. Besarnya torsi gangguan ini juga bergantung pada jarak yang ditarik dari titik rotasi roda sampai memotong tegak lurus garis kerja gaya $mg \sin \beta$ dimana besar jarak ini adalah $h - r_{\text{roda}}$.

Sehingga dapat digambarkan besar torsi gangguan ini pada (3.1).

$$\tau_{disturbance} = mgsin\beta(h - r_{roda}) \quad (3.1)$$

dengan:

$\tau_{disturbance}$	= torsi gangguan (Nm)
m	= massa kendaraan (kg)
g	= percepatan gravitasi (m/s^2)
β	= sudut kemiringan jalan ($^{\circ}$)
h	= jarak antara titik berat kendaraan dengan tanah (m)
r_{roda}	= jari-jari roda (m)

Dengan mengambil nilai parameter yaitu $m = 1200$ kg, $g = 9,81$ m/s^2 , $h=0,8$ m, $r_{roda} = 0,3$ m dan $\beta = 20^{\circ}$ untuk jalan turunan dan $\beta = -20^{\circ}$ untuk jalan tanjakan maka besar torsi gangguan untuk jalan turunan adalah sebagai berikut:

$$\tau_{disturbance} = mgsin\beta(h - r_{roda})$$

$$\tau_{disturbance} = 1200.9,81.sin(20)(0,8 - 0,3)$$

$$\tau_{disturbance} = 2013,13 Nm$$

Sedangkan besar torsi gangguan untuk jalan tanjakan adalah sebagai berikut:

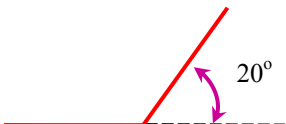

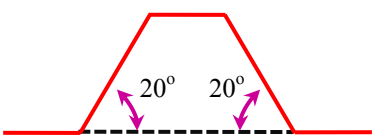
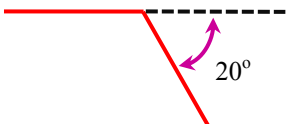
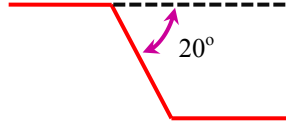
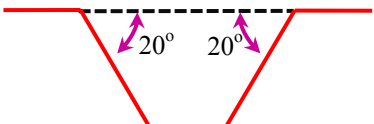
$$\tau_{disturbance} = mgsin\beta(h - r_{roda})$$

$$\tau_{disturbance} = 1200.9,81.sin(-20)(0,8 - 0,3)$$

$$\tau_{disturbance} = -2013,13 Nm$$

Pada simulasi ini model jalan dibagi menjadi 6 jenis seperti pada Tabel 3.1.

Tabel 3.1 Model Jalan

No	Model Jalan	Keterangan
1		Tanjakan akan terjadi saat detik ke-5.
2		Tanjakan akan terjadi saat detik ke-5 dan akan kembali ke jalan yang datar saat detik ke-9.
3		Tanjakan akan terjadi pada detik ke-5 dan jalan yang datar saat detik ke-9 lalu jalan akan menurun pada detik ke-11 lalu kembali lagi ke jalan yang datar pada detik ke-15.
4		Turunan akan terjadi saat detik ke-5.
5		Turunan akan terjadi saat detik ke-5 dan akan kembali ke jalan yang datar saat detik ke-9.
6		Turunan akan terjadi pada detik ke-5 dan jalan yang datar saat detik ke-9 lalu jalan akan menaik pada detik ke-11 dan lalu kembali lagi ke jalan yang datar pada detik ke-15.

3.1.2 Perancangan *Fuzzy Controller*

Dalam kajian pustaka sebelumnya dijelaskan bahwa, *fuzzy controller* dapat digunakan sebagai kontrol kecepatan pada *parallel HEV*. *Fuzzy controller* ini akan membuat kecepatan aktual PHEV sama dengan kecepatan referensi yang diberikan.

Pada penelitian ini, *fuzzy controller* akan dibuat dengan menggunakan model *plant parallel HEV* (PHEV) dari *software* MATLAB 2014a. Pada model ini, penggerak terdiri dari dua macam penggerak yaitu ICE dan *electric motor* (EM). ICE memiliki *input* yaitu *engine throttle level*, sedangkan EM memiliki *input* yaitu sinyal kontrol. Dalam perancangan *fuzzy controller* ini, ICE akan diberikan *input* yaitu *engine throttle level* yang tetap yaitu sebesar 0,1 sedangkan EM akan diberi *input* yaitu sinyal kontrol dimana nilainya berasal dari *output fuzzy controller*. Jadi *fuzzy controller* ini akan diaplikasikan pada EM pada model PHEV ini.

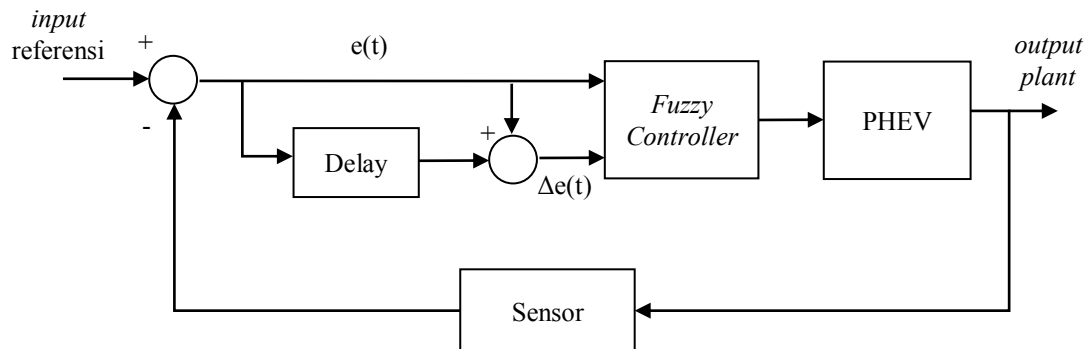
Karena *plant* yang dipakai adalah *plant* PHEV dimana terdapat operator atau pakar yang mengetahui perilaku dan karakteristik sistem yang dikendalikan, maka *fuzzy controller* yang digunakan pada penelitian ini adalah *fuzzy controller* tipe Mamdani. Pada *fuzzy controller* tipe Mamdani bagian konsekuennya berupa himpunan *fuzzy*. Jadi untuk membuat *fuzzy controller* ini harus dirancang *membership function* atau fungsi keanggotaan pada bagian *input* dan pada bagian *output* pada *fuzzy controller*.

Dalam perancangan *fuzzy controller* tipe Mamdani ini akan dipilih dua variabel keadaan sebagai anteseden aturan *fuzzy* yaitu nilai eror kecepatan angular EM (e) dan perubahan nilai eror kecepatan angular EM (Δe). Sedangkan variabel *input electric motor* pada PHEV yang merupakan variabel *output fuzzy controller* adalah torsi (u). Penghitungan sinyal eror dan perubahan nilai eror, menggunakan (3.2) dan (3.3).

$$e(t) = \omega_{\text{ref}}(t) - \omega_{\text{act}}(t) \quad (3.2)$$

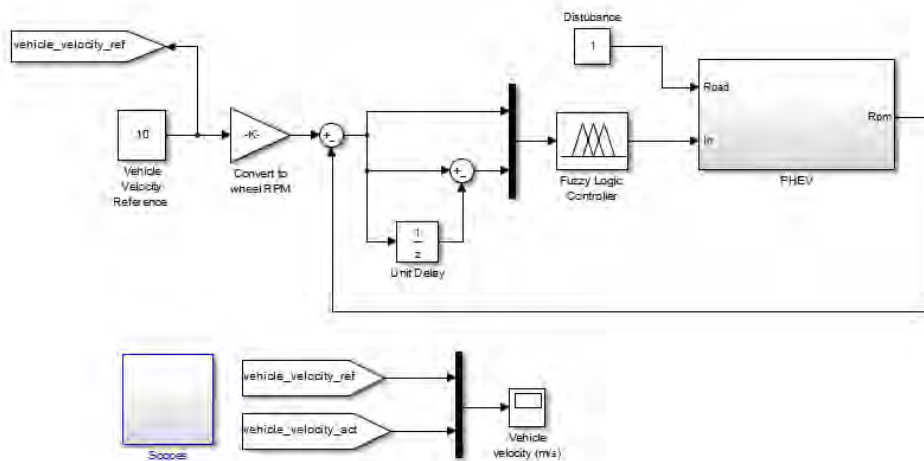
$$\Delta e(t) = e(t) - e(t - 1) \quad (3.3)$$

Sesuai dari penjelasan di atas maka struktur ini dapat digambarkan pada blok diagram seperti pada Gambar 3. 2.



Gambar 3.2 Sistem Kontrol PHEV menggunakan *Fuzzy Controller* Tipe Mamdani

Jika Gambar 3.2 dibuat simulasi pada MATLAB 2014a maka simulasi tersebut akan tampak seperti Gambar 3.3.



Gambar 3.3 Diagram *Simulink Fuzzy Controller*

Sesuai Gambar 3.3, *disturbance* dan *noise* pengukuran tidak diaplikasikan pada *model plant* PHEV terlebih dahulu. Kecepatan kendaraan akan dibuat tetap yaitu sebesar 10 m/s. Karena *fuzzy controller* ini hanya diaplikasikan pada EM dimana input dari *fuzzy controller* ini adalah kecepatan angular EM maka kecepatan kendaraan referensi harus dikonversi terlebih dahulu menjadi kecepatan angular EM. Pada Gambar 3.3 konversi ini diwakili dengan *gain* dengan nama

Convert to RPM. Pengubahan kecepatan kendaraan referensi menjadi kecepatan angular EM dihitung dengan persamaan seperti yang ditunjukkan pada (3.4).

$$\omega_{ref\ EM} = \frac{60}{2\pi \cdot r_{roda}} v_{ref\ PHEV} \quad (3.4)$$

dengan:

$\omega_{ref\ EM}$ = kecepatan angular referensi *electric motor* (rad/s)

r_{roda} = jari-jari roda (m)

$v_{ref\ PHEV}$ = kecepatan kendaraan (m/s)

Dengan memasukkan $v_{ref\ PHEV} = 10$ m/s dan $r_{roda} = 0,3$ m, maka dengan menggunakan (3.4), kecepatan angular referensi EM adalah sebagai berikut:

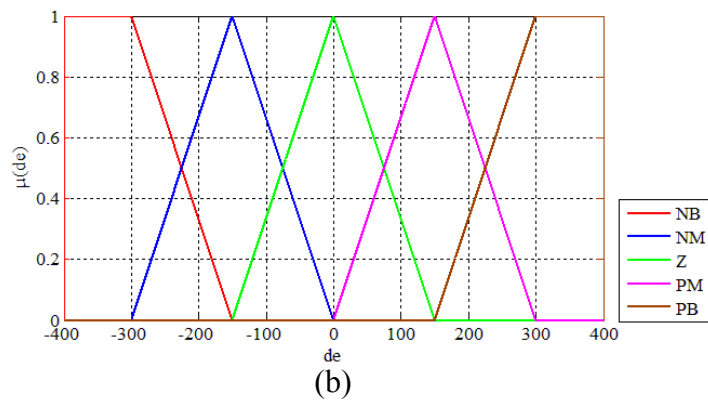
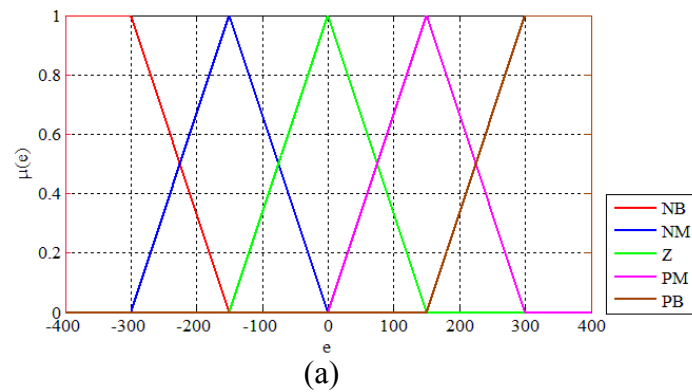
$$\omega_{ref\ EM} = \frac{60}{2\pi \cdot r_{roda}} v_{ref\ PHEV}$$

$$\omega_{ref\ EM} = \frac{60}{2\pi \cdot 0,3} 10$$

$$\omega_{ref\ EM} = 318,47\ rpm$$

Selanjutnya fungsi keanggotaan (*membership function*) *fuzzy controller* ini akan dibuat dengan tujuan agar memberikan respon kecepatan yang terbaik dimana respon kecepatan ini akan menuju kecepatan referensi yang bernilai 10 m/s.

Dari pembuatan *fuzzy controller* ini maka didapat *membership function input* seperti pada Gambar 3.4.

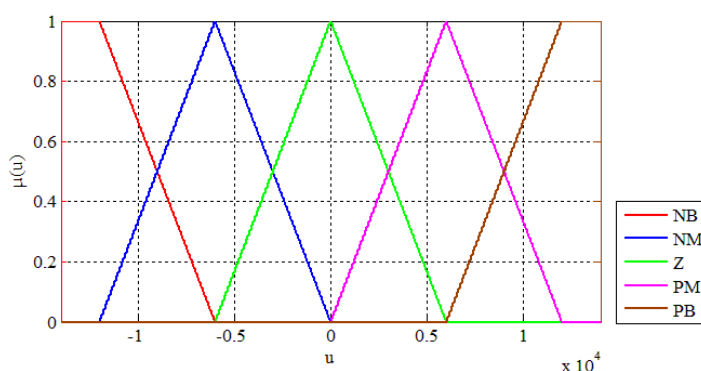


Gambar 3.4 *Membership Function Input*

(a) *e* (b) Δe

Pada fungsi keanggotaan *input* ini terdiri dari 2 variabel *fuzzy* yaitu nilai error (*e*) dan perubahan nilai error (Δe) dimana masing-masing variabel *fuzzy* ini memiliki 5 himpunan *fuzzy*, yaitu *Negative Big* (NB), *Negatif Small* (NS), *Zero* (Z), *Positive Small* (PS), *Positive Big* (PB).

Sedangkan untuk *membership function* atau fungsi keanggotaan *output*, ditunjukkan pada Gambar 3.5 di bawah ini.



Gambar 3.5 *Membership Function Output*

Pada fungsi keanggotaan *output* ini terdiri dari 1 variabel *fuzzy* yaitu sinyal kontrol (u) dimana pada variabel *fuzzy* ini juga memiliki 5 himpunan *fuzzy*, yaitu *Negative Big* (NB), *Negatif Small* (NS), *Zero* (Z), *Positive Small* (PS), *Positive Big* (PB). Pada *fuzzy controller* terdapat *rule evaluation* yang bertujuan untuk melakukan *reasoning* menggunakan nilai-nilai *fuzzy input* tersebut dan *fuzzy rule* sehingga dihasilkan *fuzzy output*. Pada pembuatan *fuzzy controller* ini dibuat *rule* sesuai yang ditunjukkan pada Tabel 3.2.

Tabel 3.2 *Rule Base Fuzzy Controller*

Δe e	NB	NM	Z	PM	PB
NB	NB	NB	NB	NM	Z
NM	NB	NB	NM	Z	PM
Z	NB	NM	Z	PM	PB
PM	NM	Z	PM	PB	PB
PB	Z	PM	PB	PB	PB

3.1.3 Perancangan *Disturbance Observer*

Pada penelitian ini, *plant parallel* HEV akan dikenakan *disturbance* yang akan menurunkan/menaikkan kecepatan aktualnya. Jadi selain adanya kontroler *fuzzy* maka diperlukan suatu metode tambahan untuk menanggulangi *disturbance* ini. *Disturbance observer* (DOB) merupakan jawaban dari permasalahan ini.

Prinsip kerja dari *disturbance observer* ini adalah mengestimasi *disturbance* yang dikenakan ke *plant parallel* HEV. Seperti pada penjelasan sebelumnya, maka *disturbance observer* ini memiliki 2 komponen utama, yaitu *invers model* dan filter yang mempunyai jenis *low pass filter*. *Invers model* yang digunakan pada *disturbance observer* penelitian ini akan menggunakan *adaptive neuro-fuzzy system* (ANFIS). Penjelasan mengenai hal ini akan dijelaskan lebih lanjut pada poin selanjutnya. Sedangkan pada perancangan *low pass filter* akan dilakukan sebagai salah satu komponen utama dalam *disturbance observer* ini.

Perancangan *low pass filter* ini merupakan salah satu langkah terpenting dalam membuat *disturbance observer* ini. Sebenarnya fungsi dari filter ini adalah untuk menekan frekuensi rendah dari *disturbance* dan frekuensi tinggi dari *noise* pengukuran. Oleh karena itu, kendala seperti frekuensi rendah dari *disturbance* dan frekuensi tinggi dari *noise* pengukuran, maka filter bertipe *low pass filter* yang cocok untuk ini. Dari Persamaan 2.29 dan 2.31 terlihat bahwa pada pelemahan *disturbance* (ξ) dikendalikan oleh parameter $1-Q(s)$, sementara pada pelemahan *noise* pengukuran dikendalikan oleh parameter $Q(s)$.

Perancangan filter pada DOB ini mengikuti struktur pada (3.5).

$$Q(s) = \frac{1 + \sum_{k=1}^{N-r} a_k(\tau s)^k}{1 + \sum_{k=1}^N a_k(\tau s)^k} \quad (3.5)$$

dengan:

N = orde dari $Q(s)$

R = orde relatif dari $Q(s)$

τ = parameter frekuensi *cutoff*

a_k = koefisien $Q(s)$

Semakin tinggi orde dari filter, maka akan semakin besar *bandwidth* dari filter untuk *me-reject disturbance*. Tetapi dengan tingginya orde dari filter maka *phase* akan menjadi jauh tertinggal. Hal ini harus menjadi pertimbangan sehingga filter ini harus memiliki orde sekecil-kecilnya agar fase tidak jauh tertinggal. Karena tujuan peneliti ini adalah membuat *plant parallel* HEV agar dapat cepat mengatasi *disturbance*, maka dibuatlah filter dengan orde 1.

Disturbance yang terjadi diasumsikan tidak akan melebihi frekuensi 2 rad/s sehingga filter akan dibuat dengan dengan frekuensi *cut-off* sebesar 2 rad/s. Dengan mengacu pada (3.5) dan dengan pemilihan koefisien filter sesuai dengan koefisien *Butterworth*, maka dapat dibuat persamaan filter untuk orde 1 seperti pada (3.6).

$$Q(s) = \frac{1 + \sum_{k=1}^{N-r} a_k (\tau s)^k}{1 + \sum_{k=1}^N a_k (\tau s)^k}$$

$$Q(s) = \frac{1 + \sum_{k=1}^{1-1} a_k (\tau s)^k}{1 + \sum_{k=1}^1 a_k (\tau s)^k}$$

$$Q(s) = \frac{1}{1 + a_k \tau s} \quad (3.6)$$

Karena koefisien *Butterworth* pada orde satu adalah $s+1$ maka nilai $a_k=1$. Sebelumnya telah ditentukan bahwa frekuensi *cutoff* yang digunakan adalah 2 rad/s. Jadi filter yang akan dipakai pada *disturbance observer* ini ditunjukkan pada (3.7) dibawah ini.

$$Q(s) = \frac{1}{1 + a_k \tau s}$$

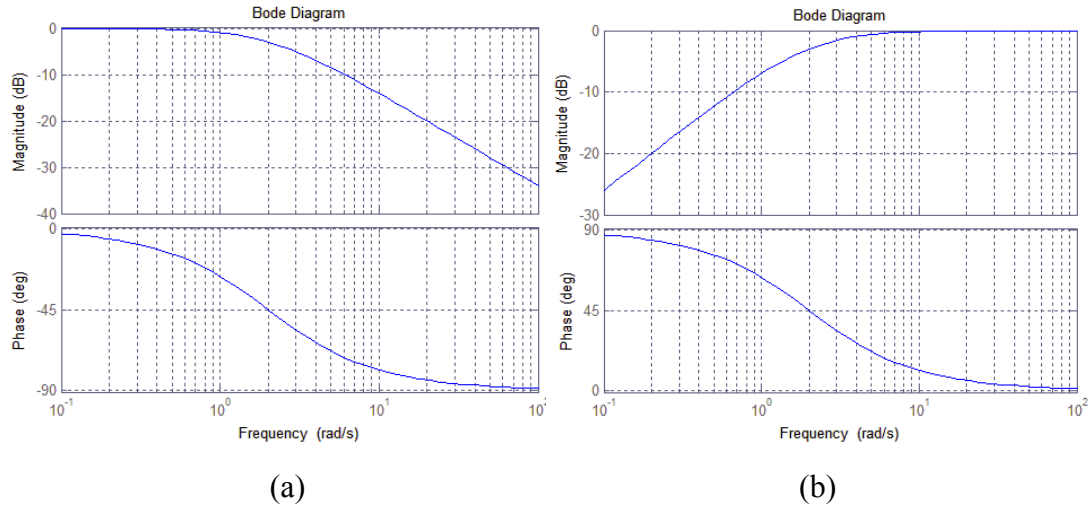
$$Q(s) = \frac{1}{1 + 1 \tau s}$$

$$Q(s) = \frac{1}{1 + \tau s}$$

$$Q(s) = \frac{1}{1 + \frac{1}{\omega_c} s}$$

$$Q(s) = \frac{1}{1 + \frac{1}{2} s} = \frac{1}{1 + 0.5s} \quad (3.7)$$

Dalam mengetahui respon *magnitude* dan *phase* dari filter pada persamaan 3.6, diagram bode filter tersebut disajikan pada Gambar 3.6.



Gambar 3.6 Diagram *Bode Filter*
(a) $Q(s)$ (b) $I-Q(s)$

Pada Gambar 3.6 (a) terlihat bahwa *noise* pengukuran yang terjadi di atas frekuensi 2 rad/s akan diredam. Pada frekuensi 100 rad/s, filter ini akan meredam sampai -34 dB dengan fase yang tertinggal sebesar 90 derajat. Sedangkan pada Gambar 3.6(b) terlihat bahwa *disturbance* akan teredam pada frekuensi dibawah frekuensi 2 rad/s. Pada frekuensi 0,1 rad/s, filter ini akan meredam sampai -26 dB, dengan fase yang mendahului sebesar 90 derajat.

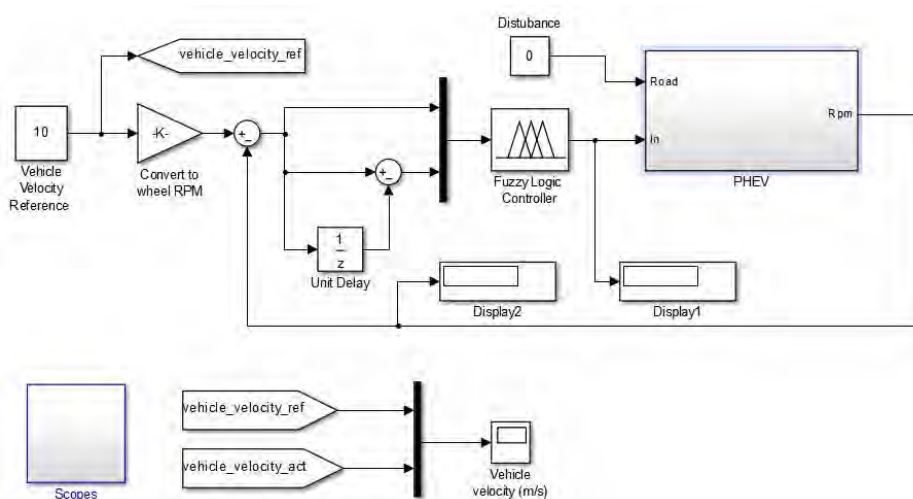
3.1.4 Perancangan *Neuro-Fuzzy Invers Model*

Adaptive Neuro-Fuzzy System (ANFIS) adalah sistem yang merupakan gabungan antara *fuzzy logic* dengan jaringan saraf tiruan. Jika dibandingkan dengan jaringan saraf tiruan maka ANFIS memiliki *neuron* berupa *fuzzy* sedangkan pada jaringan saraf tiruan berupa bobot. Sedangkan jika dibandingkan dengan *fuzzy* maka ANFIS membentuk *rule* dengan proses pembelajaran seperti pada jaringan saraf tiruan sedangkan pada *fuzzy* penyusunan *rule* tidak melalui proses pembelajaran ini. Proses pembelajaran ini akan dilakukan terus menerus sampai selisih *output* dari ANFIS dengan data *output* sebenarnya sebesar

bilangan tertentu. Selain itu dapat juga pelatihan ANFIS dilakukan sampai beberapa kali dimana banyaknya pelatihan dibatasi oleh bilangan tertentu.

Salah satu keuntungan ANFIS adalah ANFIS dapat membuat hubungan antara data *input* dan *output* tanpa mengetahui persamaan yang menghubungkan data *input* dan *output*nya. Karena pada tesis ini, *invers model parallel Hybrid Electric Vehicle* (HEV) dibutuhkan sebagai salah satu komponen dalam *disturbance observer*, maka ANFIS ini akan digunakan untuk membuat *invers model parallel Hybrid Electric Vehicle* (HEV).

Ada beberapa tahapan yang dilakukan dalam membuat *neuro-fuzzy invers model*. Pertama, proses pencarian data pasangan *input output* harus dilakukan. Karena ANFIS digunakan sebagai *invers model* maka data *input* ANFIS berupa data *output* PHEV sedangkan data *output* ANFIS berupa data *input* PHEV. Dalam memperoleh data *input output* ini, dibutuhkan simulasi seperti pada Gambar 3.7.



Gambar 3.7 Diagram Simulink Pencarian Data Training ANFIS

Pada Gambar 3.7, *display 2* merupakan data *input* sedangkan *display 1* merupakan data *output* sehingga jika dikumpulkan maka didapat pasangan *input* dan *output* dari *parallel HEV*. Untuk mendapatkan data tersebut, *vehicle velocity reference* diubah dari 0 m/s sampai 10 m/s sehingga akan tampil pasangan data pada *display 1* dan *display 2*. Data-data inilah yang menjadi data untuk pelatihan ANFIS.

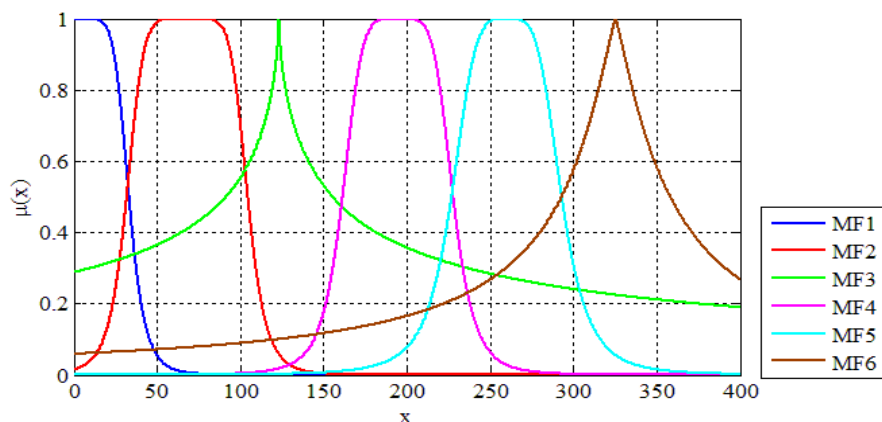
Setelah mendapatkan data pasangan *input output* untuk pembelajaran struktur ANFIS maka tahap selanjutnya adalah melakukan proses *training* pada struktur ANFIS tersebut. Jaringan ANFIS yang dipakai menggunakan 6 *membership function input* yang bertipe *generalized bell-shaped* dan *membership function output* bertipe *linier*. Pada ANFIS, *training* dilakukan dengan menggunakan *anfisedit* pada MATLAB 2014a. *Training* ini dilakukan untuk mencari *premise parameters*, dan *consequent parameters*.

Dari hasil *training* yang dilakukan, maka didapat hasil *premise parameters* untuk keenam *membership function* (MF) yang bertipe *generalized bell-shaped* seperti pada Tabel 3.3.

Tabel 3.3 *Premise Parameters*

MF \ Parameter	a	b	c
1	32,86	3,451	0,4372
2	36,38	3,367	68,22
3	32,17	0,3371	123,3
4	33,23	2,619	194,7
5	33,1	2,075	260
6	32,69	0,6021	325,4

Jika digambarkan *membership function input* ANFIS bertipe *generalized bell-shaped* di atas dapat dilihat pada Gambar 3.8.



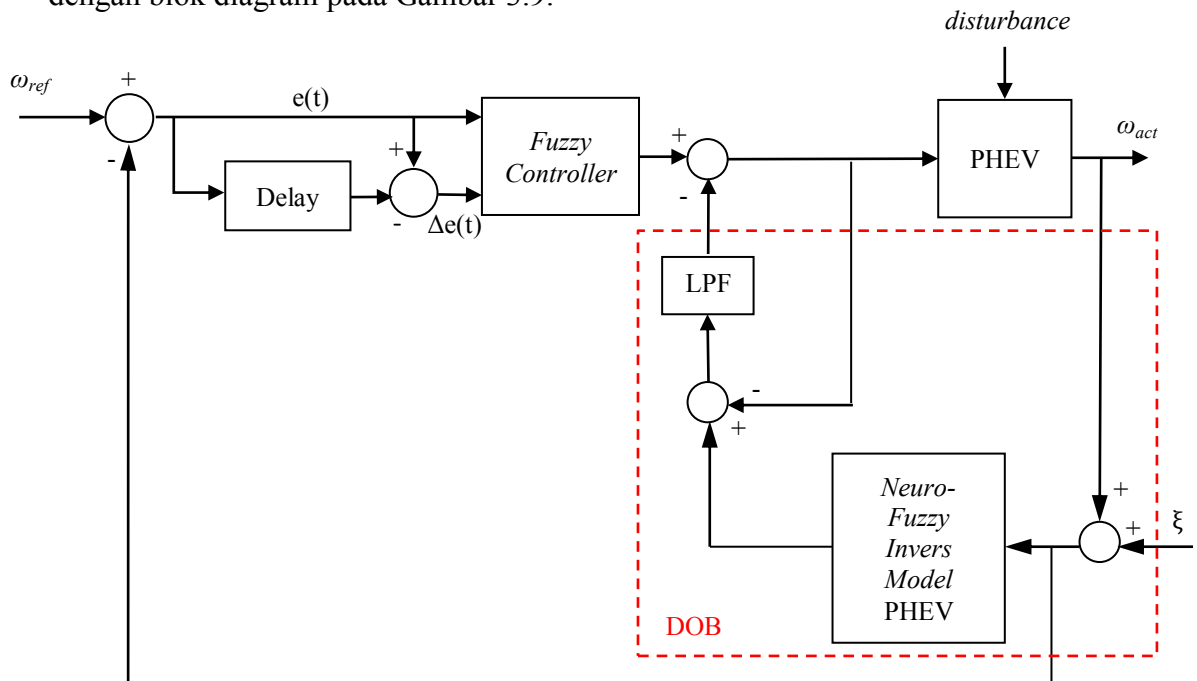
Gambar 3.8 *Membership Function Input ANFIS*

Sedangkan hasil dari *consequent parameters* dapat dilihat pada Tabel 3.4 dibawah ini.

Tabel 3.4 *Consequent Parameters*

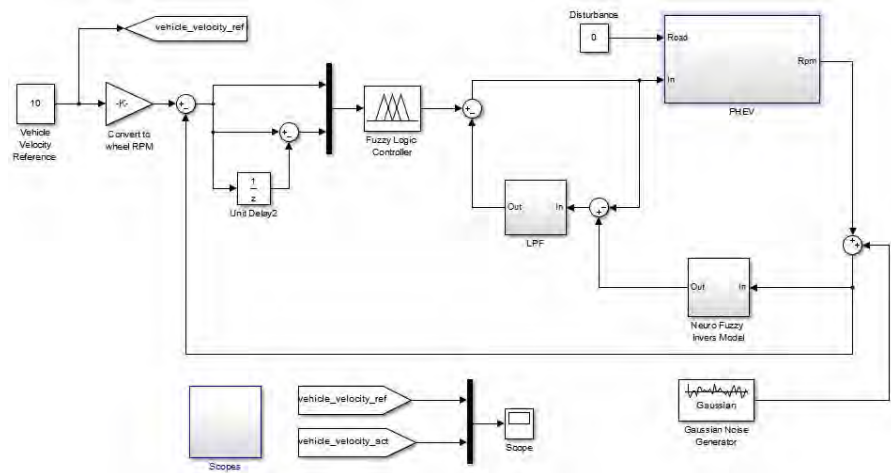
Parameter MF	c_1	c_2
1	1,11	-128
2	0,9031	-140,3
3	-2,638	190,4
4	0,154	-51,81
5	0,05891	-27,96
6	-3,472	1253

Sesuai pada Tabel 3.3 dan 3.4 maka parameter-parameter ini akan membentuk ANFIS menjadi *neuro-fuzzy invers model* pada model *parallel HEV* di MATLAB 2014a. Setelah semua perancangan dilakukan, maka *fuzzy controller* dan *disturbance observer* dengan *neuro-fuzzy invers model* dapat dibentuk sesuai dengan blok diagram pada Gambar 3.9.



Gambar 3.9 Blok Diagram *Fuzzy Disturbance Observer*

Seperti terlihat pada blok diagram pada Gambar 3.9, maka dapat dibentuk diagram simulink *fuzzy disturbance observer* pada MATLAB 2014a. Diagram *simulink* tersebut diperlihatkan pada Gambar 3.10.



Gambar 3.10 Diagram *Simulink Fuzzy Disturbance Observer*

BAB IV

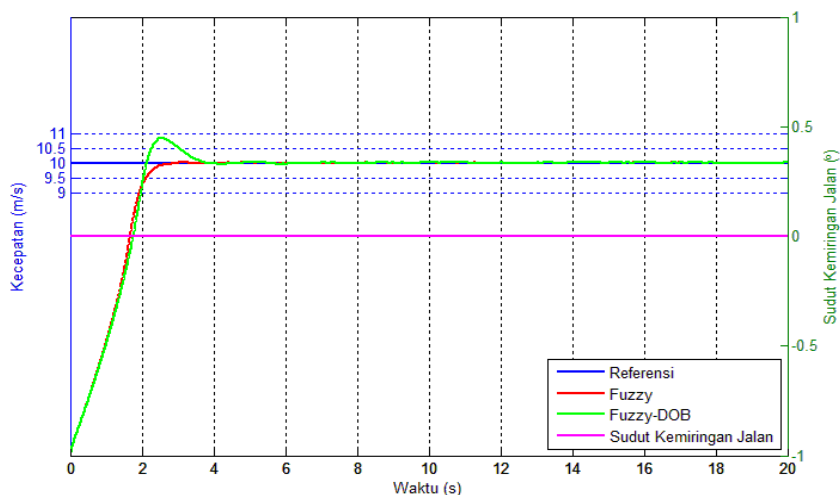
HASIL DAN PEMBAHASAN

4.1 Gambaran Umum Pengujian Sistem

Pengujian pada penelitian ini akan difokuskan terhadap simulasi menggunakan MATLAB R2014a dengan *plant* yang berupa *Parallel Hybrid Electric Vehicle* (PHEV). Pengujian ini akan dilakukan dengan melihat ketahanan sistem terhadap pemberian *disturbance* dan dengan melihat respon sistemnya. Pengujian dilakukan dengan menggunakan 2 jenis kontrol yaitu kontroler *fuzzy* dan *fuzzy disturbance observer* dengan menggunakan *neuro-fuzzy invers model*. Semua pengujian akan dilakukan dengan menambahkan *noise* pengukuran dimana memiliki parameter yaitu *mean value* adalah 0, nilai *variance* adalah 10, *initial seed* sebesar 41 dan *sample time* sebesar 0,01.

4.1.1 Pengujian tanpa *Disturbance*

Pada pengujian ini, PHEV akan disimulasikan dimana jalan yang dilalui oleh PHEV adalah jalan yang datar atau tidak memiliki kemiringan. Karena jalan tersebut tidak memiliki kemiringan atau dengan kata lain memiliki kemiringan 0° , maka PHEV tidak terkena gangguan atau *disturbance*. Pada pengujian ini nilai referensi kecepatan kendaraan adalah sebesar 10 m/s. Hasil dari simulasi ini tampak seperti pada Gambar 4.1.



Gambar 4.1 Respon Sistem Pengujian tanpa *Disturbance*

Respon pada pengujian PHEV pada jalan yang datar seperti yang terlihat pada Gambar 4.1, memberikan hasil bahwa kedua kontroler dapat mengikuti kecepatan referensi. Pada respon *fuzzy disturbance observer*, PHEV berjalan selama 20 detik diatas jalan sepanjang 189 m, dengan kecepatan akhir sebesar 10 m/s. Sedangkan pada respon kontroler fuzzy, PHEV berjalan selama 20 detik diatas jalan sepanjang 188,5 m dengan kecepatan akhir yang sama yaitu 10 m/s. Tetapi perbedaan antara kedua kontroler tersebut terletak pada *overshoot*, *rise time*, dan *settling time*. Kontroler *fuzzy disturbance observer* cenderung memiliki *overshoot* pada hasil simulasinya. *Overshoot* pada *fuzzy disturbance observer* yaitu sebesar 10,85 m/s. Kemudian untuk *rise time* pada *fuzzy disturbance observer* yaitu sebesar 1,98 detik dan *settling time* yaitu sebesar 3,85 detik. Sedangkan kontroler *fuzzy* memberikan *overshoot*, *rise time*, dan *settling time* yang berbeda. Pada kontroler *fuzzy*, hasil simulasi menunjukkan bahwa tidak ada *overshoot*, *rise time* yaitu sebesar 1,94 detik dan *settling time* yaitu sebesar 2,95 detik.

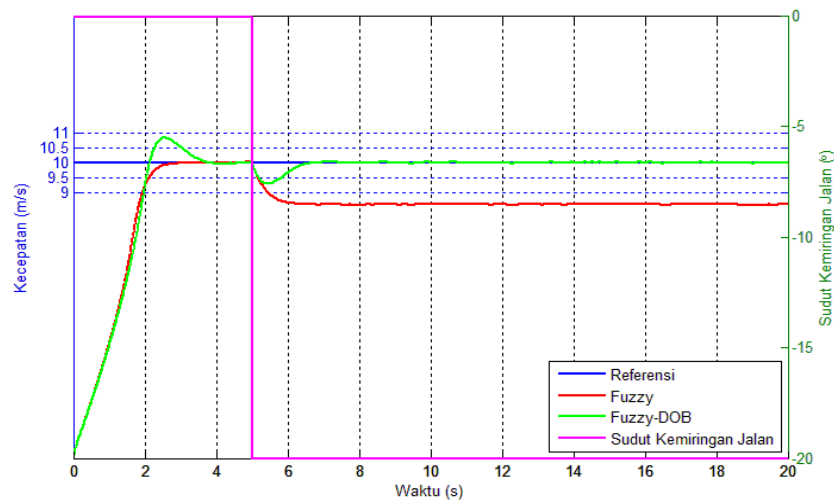
Dari sini terlihat bahwa jika dibandingkan dengan kontroler *fuzzy*, maka kontroler *fuzzy disturbance observer* menimbulkan efek *overshoot* pada respon sistem dan menambah lama waktu *rise time* dan *settling time*.

4.1.2 Pengujian dengan *Disturbance* I

Pada pengujian ini, akan dibuat simulasi PHEV yang melalui jalan tanjakan dengan kemiringan 20° . Karena jalan yang dilalui oleh PHEV ini berupa tanjakan yang memiliki kemiringan 20° maka PHEV akan terkena gangguan atau *disturbance* yang akan mengurangi kecepatannya. Model jalan yang dilalui PHEV ini digambarkan pada *disturbance* nomor 1 di Tabel 3.1.

Disturbance nomor 1 ini memiliki kemiringan jalan tanjakan sebesar 20° pada detik ke-5 dan begitu seterusnya. Nilai dari kecepatan referensi kendaraan pada pengujian ini mempunyai nilai yang sama dengan pengujian sebelumnya yaitu sebesar 10 m/s.

Respon sistem dari hasil pengujian ini ditunjukkan pada Gambar 4.2.



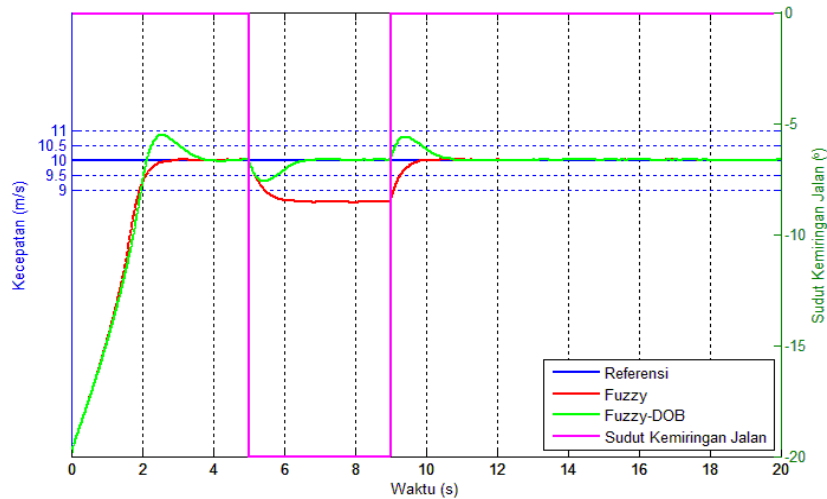
Gambar 4.2 Respon Sistem Pengujian dengan *Disturbance I*

Respon sistem yang diperlihatkan pada Gambar 4.2, masing-masing kontroler memberikan kecepatan aktual akhir yang berbeda. Pada kontroler *fuzzy disturbance observer*, setelah melalui jalan lurus sepanjang 38,9 m, kecepatan aktualnya akan turun menuju 9,3 m/s tetapi, 1,75 detik kemudian kecepatan aktualnya menjadi 10 m/s. Jadi setelah berjalan melalui tanjakan sepanjang 16,9 m maka PHEV akan kembali ke kecepatan 10 m/s. Kecepatan ini akan tetap 10 m/s sampai detik ke-20 atau selama PHEV bergerak sejauh 149,6 meter diatas tanjakan. Hal ini berbeda dengan respon kontroler *fuzzy*. Pada respon kontroler *fuzzy*, PHEV yang berjalan di atas tanjakan selama 15 detik sepanjang 129,4 m tetap tidak bisa membuat PHEV menuju ke kecepatan referensi sebesar 10 m/s. Setelah berjalan melalui jalan lurus sepanjang 38,3 m kecepatan aktual PHEV yang menggunakan kontroler *fuzzy* ini akan turun sebesar 8,6 m/s. Jadi *disturbance* nomor 1 ini memberikan pengaruh eror kecepatan sebesar 1,4 m/s ketika menggunakan kontroler *fuzzy*.

4.1.3 Pengujian dengan *Disturbance II*

Pada pengujian ini, akan disimulasikan PHEV dimana jalan yang dilalui PHEV seperti pada model jalan pada *disturbance* nomor 2 seperti pada Tabel 3.1. Model jalan yang dilalui PHEV ini memiliki kemiringan jalan tanjakan sebesar

20° pada detik ke-5 dan akan kembali ke jalan yang datar atau kemiringan sebesar 0° saat detik ke-9. Nilai dari kecepatan referensi kendaraan pada pengujian ini mempunyai nilai yang sama dengan pengujian sebelumnya yaitu sebesar 10 m/s. Respon sistem dari hasil pengujian ini ditunjukkan pada Gambar 4.3.



Gambar 4.3 Respon Sistem Pengujian dengan *Disturbance II*

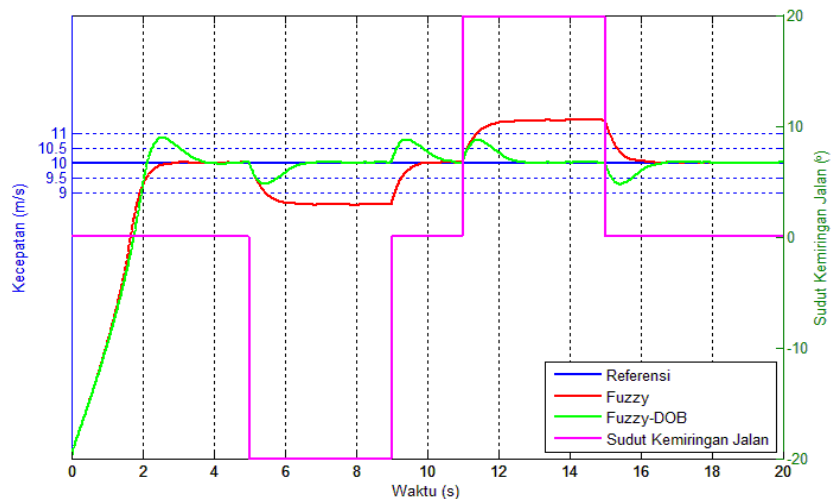
Pada *fuzzy disturbance observer*, setelah bergerak sejauh 38,9 m pada jalan lurus, kecepatan aktual mempunyai respon yang mendekati kecepatan referensi yaitu 10 m/s, 1,75 detik setelah terjadinya *disturbance* pada detik-5. Jadi selama bergerak sejauh 16,9 m, kecepatan aktual sempat turun sebesar 9,3 m/s. Tetapi pada detik ke-9, atau setelah berjalan sejauh 39,4 m diatas tanjakan, kecepatan aktual akan naik menjadi 10,8 m/s dan kemudian turun sampai sebesar 10 m/s pada 1,75 detik kemudian atau bergerak sejauh 18,2 m. Kecepatan ini akan tetap selama bergerak diatas jalan lurus sepanjang 110,8 m pada detik ke-20.

Hasil respon pada Gambar 4.3 juga memperlihatkan bahwa pada kontroler *fuzzy*, setelah bergerak pada jalan lurus sejauh 38,3 m, kecepatan aktual turun pada detik ke-5 dan naik kembali pada detik ke-9 menuju 10 m/s sampai pada detik ke-20 atau selama bergerak di atas jalan lurus sejauh 109,8 m. Penurunan kecepatan ini adalah sebesar 8,6 m/s selama bergerak sejauh 34,9 m diatas tanjakan. Jadi dapat dilihat bahwa, selama gangguan atau *disturbance* dikarenakan akibat jalan tanjakan tidak hilang, maka pada *fuzzy controller* kecepatan aktual tidak akan kembali ke 10 m/s.

4.1.4 Pengujian dengan *Disturbance III*

Pada pengujian ini, PHEV akan disimulasikan dengan menggunakan model jalan seperti dengan *disturbance* nomor 3 seperti pada Tabel 3.1. *Disturbance* nomor 3 ini memiliki kemiringan jalan tanjakan sebesar 20° pada detik ke-5 dan kembali ke jalan yang datar saat detik ke-9 kemudian jalan akan menurun pada detik ke-11 dan kemudian kembali lagi ke jalan yang datar pada detik ke-15. Nilai dari kecepatan referensi juga 10 m/s.

Hasil respon yang dihasilkan oleh kontroler *fuzzy* dan *fuzzy disturbance observer* ini ditunjukkan pada Gambar 4.4.



Gambar 4.4 Respon Sistem Pengujian dengan *Disturbance III*

Hasil respon pada Gambar 4.4 terlihat bahwa pada *fuzzy disturbance observer*, setelah bergerak ada jalan lurus sepanjang 38,9 m, kecepatan aktual mempunyai respon yang mendekati kecepatan referensi, tetapi akibat tanjakan pada detik ke-5, kecepatan aktual akan turun sampai sebesar 9,3 m/s. Setelah 1,75 detik atau setelah bergerak sejauh 16,9 m kemudian, kecepatan aktual menjadi 10 m/s. Sesudah bergerak sejauh 39,4 m diatas tanjakan, jalan akan kembali lurus pada detik ke-9, maka kecepatan aktual naik sebesar 10,8 m/s dan kembali menjadi 10 m/s setelah bergerak sejauh 18,2 m selama 1,75 detik. Jalan lurus ini membentang sejauh 20,7 meter.

Pada detik ke-11, jalan kembali menurun, sehingga kecepatan naik sebesar 10,8 m/s kemudian menurun sebesar 10 m/s setelah bergerak sejauh 18,1 m selama 1,75 detik. Jalan turunan ini dilewati oleh PHEV sejauh 40,7 m. Pada detik

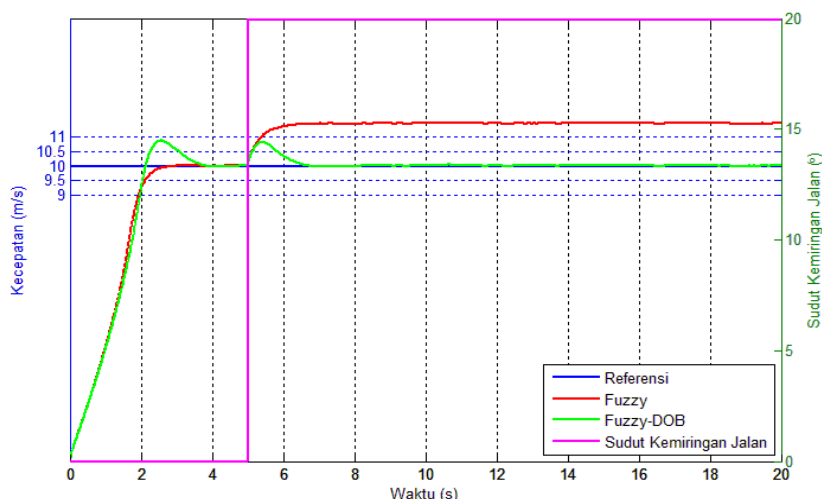
ke-15, jalan kembali lurus. Hal ini membuat kecepatan turun sebesar 9,3 m/s yang kemudian naik menjadi 10 m/s setelah bergerak selama 1,75 detik sejauh 16,9 m. Kecepatan akhir 10 m/s ini tetap konstan selama jalan lurus sampai detik ke-20, sejauh 49,5 m.

Hal ini berbeda dengan kontroler *fuzzy*, dimana setelah bergerak diatas jalan yang lurus sepanjang 38,3 m, terdapat jalan tanjakan sepanjang 34,9 m, kecepatan turun menjadi 8,6 m/s, dan hanya menuju ke kecepatan 10 m/s jika jalan kembali lurus pada detik ke-9 sampai ke detik ke-11 sepanjang 19,7 m. Sedangkan pada detik ke-11, terdapat jalan turunan sampai detik ke-15 sepanjang 45,9 m, maka akan membuat kecepatan naik sebesar 11,44 m/s. Dan akan kembali ke 10 m/s ketika jalan kembali lurus pada detik ke-15 sampai detik ke-20 yaitu sepanjang 50,9 m.

4.1.5 Pengujian dengan *Disturbance* IV

Pada pengujian ini, PHEV akan disimulasikan dengan membuat PHEV melalui jalan turunan dengan kemiringan 20° . Karena jalan yang dilalui oleh PHEV ini berupa turunan yang memiliki kemiringan 20° maka PHEV akan terkena gangguan atau *disturbance* yang akan menambah kecepatannya. Model jalan yang dilalui PHEV ini digambarkan pada *disturbance* nomor 4 di Tabel 3.1. *Disturbance* nomor 4 ini memiliki kemiringan jalan turunan sebesar 20° pada detik ke-5 dan begitu seterusnya. Nilai dari kecepatan referensi kendaraan pada yaitu sebesar 10 m/s.

Hasil respon yang dihasilkan oleh kontroler *fuzzy* dan *fuzzy disturbance observer* ini ditunjukkan pada Gambar 4.5.



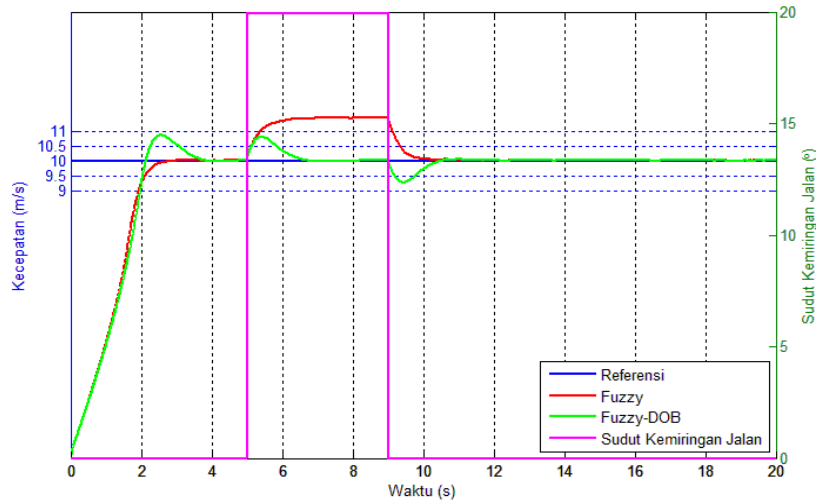
Gambar 4.5 Respon Sistem Pengujian dengan *Disturbance IV*

Respon sistem yang diperlihatkan pada Gambar 4.5, masing-masing kontroler memberikan kecepatan aktual akhir yang berbeda. Pada kontroler *fuzzy disturbance observer*, setelah melalui jalan lurus sepanjang 38,9 m, kecepatan aktualnya akan naik menuju 10,8 m/s tetapi 1,75 detik kemudian, kecepatan aktualnya menjadi 10 m/s. Jadi setelah berjalan melalui turunan sepanjang 18,2 m maka PHEV akan kembali ke kecepatan 10 m/s. Kecepatan ini akan tetap 10 m/s sampai detik ke-20 atau selama PHEV bergerak sejauh 150,8 meter diatas turunan. Hal ini berbeda dengan respon kontroler *fuzzy*. Pada respon kontroler *fuzzy*, PHEV yang berjalan di atas turunan selama 15 detik sepanjang 171,1 m tetap tidak bisa membuat PHEV menuju ke kecepatan referensi sebesar 10 m/s. Setelah berjalan melalui jalan lurus sepanjang 38,3 m, kecepatan aktual PHEV yang menggunakan kontroler *fuzzy* ini akan naik sebesar 11,44 m/s. Jadi *disturbance* nomor 4 ini memberikan pengaruh eror kecepatan sebesar 1,44 m/s ketika menggunakan kontroler *fuzzy*.

4.1.6 Pengujian dengan *Disturbance V*

Pengujian ini akan disimulasikan PHEV dimana jalan yang dilalui PHEV seperti pada model jalan pada *disturbance* nomor 5 seperti pada Tabel 3.1. *Disturbance* nomor 5 ini memiliki kemiringan jalan turunan sebesar 20° pada

detik ke-5 dan akan kembali ke jalan yang datar dimana memiliki kemiringan 0° saat detik ke-9. Sedangkan nilai dari kecepatan referensi kendaraan pada pengujian ini yaitu sebesar 10 m/s. Respon sistem dari hasil pengujian ini ditunjukkan pada Gambar 4.6.



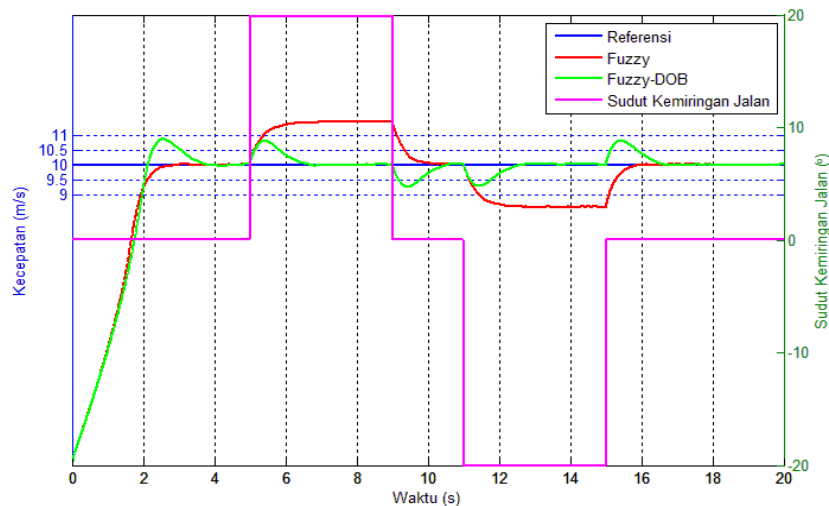
Gambar 4.6 Respon Sistem Pengujian dengan *Disturbance V*

Pada *fuzzy disturbance observer*, setelah bergerak sejauh 38,9 m pada jalan lurus, kecepatan aktual mempunyai respon yang mendekati kecepatan referensi yaitu 10 m/s, 1,75 detik setelah terjadinya *disturbance* pada detik-5. Jadi selama bergerak sejauh 18,2 m, kecepatan aktual sempat naik sebesar 10,8 m/s. Tetapi pada detik ke-9, atau setelah berjalan sejauh 40,7 m diatas turunan, kecepatan aktual akan turun menjadi 9,3 m/s dan kemudian naik sampai sebesar 10 m/s pada 1,75 detik kemudian atau bergerak sejauh 16,9 m. Kecepatan ini akan tetap selama bergerak diatas jalan lurus sepanjang 109,5 m pada detik ke-20.

Hasil respon pada Gambar 4.6 juga memperlihatkan bahwa pada kontroler *fuzzy*, setelah bergerak pada jalan lurus sejauh 38,3 m, kecepatan aktual naik pada detik ke-5 dan turun kembali pada detik ke-9 menuju 10 m/s sampai pada detik ke-20 atau selama bergerak di atas jalan lurus sejauh 110,6 m. Kenaikan kecepatan ini adalah sebesar 11,44 m/s selama bergerak sejauh 45,3 m diatas turunan. Jadi dapat dilihat bahwa, selama gangguan atau *disturbance* dikarenakan akibat jalan turunan tidak hilang, maka pada *fuzzy controller* kecepatan aktual tidak akan kembali ke 10 m/s.

4.1.8 Pengujian dengan *Disturbance VI*

Pada pengujian ini, PHEV akan disimulasi dengan *disturbance* nomor 6 seperti pada Tabel 3.1. *Disturbance* nomor 6 ini memiliki kemiringan jalan turunan sebesar 20° pada detik ke-5 dan kembali ke jalan yang datar saat detik ke-9 kemudian jalan akan menaik pada detik ke-11 dan kemudian kembali lagi ke jalan yang datar pada detik ke-15. Ketika diberikan kecepatan referensi sebesar 10 m/s, maka akan dihasilkan respon seperti yang ditunjukkan pada Gambar 4.7.



Gambar 4.7 Respon Sistem Pengujian dengan *Disturbance VI*

Hasil respon pada Gambar 4.7 terlihat bahwa pada *fuzzy disturbance observer*, setelah bergerak ada jalan lurus sepanjang 38,9 m, kecepatan aktual mempunyai respon yang mendekati kecepatan referensi, tetapi akibat turunan pada detik ke-5, kecepatan aktual akan naik sampai sebesar 11,44 m/s. Setelah 1,75 detik atau setelah bergerak sejauh 18,2 m kemudian, kecepatan aktual menjadi 10 m/s. Sesudah bergerak sejauh 40,7 m diatas turunan, jalan akan kembali lurus pada detik ke-9, maka kecepatan aktual turun sebesar 9,3 m/s dan kembali menjadi 10 m/s setelah bergerak sejauh 16,9 m selama 1,75 detik. Jalan lurus ini membentang sejauh 19,4 meter.

Pada detik ke-11, jalan kembali menaik, sehingga kecepatan turun sebesar 9,3 m/s kemudian naik sebesar 10 m/s setelah bergerak sejauh 16,9 m selama 1,75 detik. Jalan tanjakan ini dilewati oleh PHEV sejauh 39,5 m. Pada detik ke-15, jalan kembali lurus. Hal ini membuat kecepatan naik sebesar 10,8 m/s yang kemudian turun menjadi 10 m/s setelah bergerak selama 1,75 detik sejauh 18,2 m.

Kecepatan akhir 10 m/s ini tetap konstan selama jalan lurus sampai detik ke-20, sejauh 50,7 m.

Hal ini berbeda dengan kontroler *fuzzy*, dimana setelah bergerak diatas jalan yang lurus sepanjang 38,3 m, terdapat jalan turunan sepanjang 45,3 m, kecepatan naik menjadi 11,4 m/s, dan hanya menuju ke kecepatan 10 m/s jika jalan kembali lurus pada detik ke-9 sampai ke detik ke-11 sepanjang 20,5 m. Sedangkan pada detik ke-11, terdapat jalan tanjakan sampai detik ke-15 sepanjang 34,9 m, maka akan membuat kecepatan turun sebesar 8,6 m/s. Dan akan kembali ke 10 m/s ketika jalan kembali lurus pada detik ke-15 sampai detik ke-20 yaitu sepanjang 49,7 m.

Dari pengujian-pengujian yang telah dilakukan maka dapat dirangkum *steady state error* pada tiap kontroler yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 *Steady State Error*

Kontroler	Sudut Kemiringan Jalan		
	0°	20°	-20°
<i>Fuzzy</i>	0,05 m/s	1,4 m/s	-1,4 m/s
<i>Fuzzy Disturbance Observer</i>	0,05 m/s	0,05 m/s	0,05 m/s

BAB V

PENUTUP

5.1 Kesimpulan

Pada penelitian ini, telah dilakukan pengujian terhadap dua kontroler yaitu kontroler *fuzzy*, dan *fuzzy disturbance observer* pada model *plant parallel hybrid electric vehicle* (PHEV) dari MATLAB 2014a. Pengujian ini dilakukan dengan simulasi MATLAB 2014a dimana model *plant* PHEV akan dikenai gangguan (*disturbance*) yang diakibatkan oleh enam jenis model jalan dimana tiap model jalan memiliki kemiringan sebesar 20° .

Hasil simulasi menunjukkan bahwa kontroler *fuzzy disturbance observer* akan membuat kecepatan aktual sama dengan kecepatan referensi yang nilainya sebesar 10 m/s. Ketika PHEV mengalami gangguan maka kecepatan aktualnya akan turun sebesar 9,3 m/s jika PHEV melalui tanjakan dan naik sebesar 10,8 m/s ketika PHEV melalui jalan turunan. Tetapi walaupun terjadi perbedaan antara kecepatan aktual dengan kecepatan referensi, 1,75 detik setelah terjadi gangguan, kecepatan aktual akan menuju 10 m/s.

Hasil simulasi yang berbeda ditunjukkan ketika kontroler *fuzzy* yang digunakan pada model PHEV. Kontroler *fuzzy* akan membuat kecepatan aktual turun menjadi 8,6 m/s ketika PHEV melalui jalan tanjakan dan naik menjadi 11,44 m/s ketika PHEV melalui jalan turunan. Kontroler *fuzzy* akan membuat kecepatan aktual menuju kecepatan referensi hanya ketika tidak ada gangguan yang dialami oleh PHEV. Jadi ketika PHEV mengalami gangguan, kecepatan aktual tidak akan sama dengan kecepatan referensi.

Dari hasil ini, dapat diambil kesimpulan yaitu *fuzzy disturbance observer* dapat diaplikasikan pada *plant parallel HEV* dimana kontroler ini dapat menolak *disturbance* yang diakibatkan dari bentuk jalan dengan kemiringan tertentu.

5.2 Saran

Pada penelitian mengenai penggunaan kontroler *fuzzy disturbance observer* dengan *neuro-fuzzy invers model* ini, PHEV akan diberi 1 jenis *disturbance* yaitu *disturbance* yang diakibatkan oleh jalan yang dilalui model PHEV memiliki kemiringan tertentu. Pada pengembangan penelitian selanjutnya, dapat digunakan *disturbance* tidak hanya dikarenakan pada kemiringan jalan tetapi dapat ditambah dengan jenis *disturbance* yang lain.

DAFTAR PUSTAKA

- [1] S. Wahsh, H. G. Hamed, M. N. F. Nashed, and T. Dakroy, "Fuzzy Logic Based Control Strategy for Parallel Hybrid Electric Vehicle", *Proceedings of 2008 IEEE International Conference on Mechatronics and Automation*, Pages:27 – 31, 20 May 2008.
- [2] Mariam Khan and Narayan C.Kar, "Speed Tracking Performance of Fuzzy Based Vector Control of Induction Motor Drives for Hybrid Electric Vehicles", *Canadian Conference on Electrical and Computer Engineering*, Pages:000607– 000610, 5 May 2008.
- [3] A.M. Harsha S. Abeykoon, Hasala R. Senevirathne, "Disturbance Observer Based Current Controller for a DC motor", *International Conference on Information and Automation for Sustainability*, Pages:47– 52, 2012.
- [4] J.C. Gonzalez-Gomez, J.A. Ruz-Hernandez, R. Garcia-Hernandez, and E.N. Sanchez, "Real-Time Neuro-Fuzzy Inverse Control Applied to A DC Motor", *Western Athletic Conference*, Pages:1– 5, 2011.
- [5] M. Habib Ullah, T.S. Gunawan, and R. Muhida, "Design of Environmental Friendly Hybrid Electric Vehicle" *International Conference on Computer and Communication Engineering*, Pages:544-548, 2012.
- [6] Aymeric Rousseau, "Hybrid Electric Vehicles", 12 September 2014, http://www.autonomie.net/references/hev_26c.html.
- [7] Mathworks Corporate Headquarters, "Parallel Hybrid Transmission, 12 Desember 2014, <http://www.mathworks.com/help/phymod/sdl/examples/parallel-hybrid-transmission.html>.
- [8] Chun-Chih Wang and Masayoshi Tomizuka, "Design of Robustly Stable Disturbance Observers Based on Closed Loop Consideration Using H_∞ Optimization and Its Applications to Motion Control Systems", *Proceedings of 2004 American Control Conference*, Pages:3764-3769, 2004.
- [9] Dwi Ana Ratna Wati, "Sistem Kendali Cerdas:Fuzzy Logic Controller (FLC), Jaringan Syaraf Tiruan (JST), Algoritma Genetik (AG), dan Algoritma Particle Swarm Optimization (PSO)", 1th edition, Graha Ilmu, Yogyakarta, 2011.

- [10] Suyanto, “*Soft Computing Membangun Mesin Ber-IQ Tinggi*”, 1nd edition, Penerbit Informatika, Bandung, 2008.

BIOGRAFI PENULIS



Penulis bernama lengkap Yoga Alif Kurnia Utama, lahir pada tanggal 11 April 1990 di Mojokerto. Penulis merupakan anak pertama dari pasangan Ali Fikri dan Pancawati. Penulis telah melalui serangkaian pendidikan formal mulai tahun 1996 sampai 2015. Pendidikan Formal yang ditempuh oleh penulis adalah sebagai berikut:

- Tahun 2002 lulus dari SD Taruna Nusa Harapan, Mojokerto
- Tahun 2005 lulus dari SLTP Taruna Nusa Harapan, Mojokerto
- Tahun 2008 lulus dari SMA ST. Louis I, Surabaya
- Tahun 2012 lulus dari Jurusan Teknik Mekatronika, Politeknik Elektronika Negeri Surabaya
- Tahun 2013 tercatat sebagai mahasiswa Bidang Keahlian Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya

Pada tanggal 28 Juli 2015, penulis telah mempertanggungjawabkan tesis dalam ujian tesis di Bidang Keahlian Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

DAFTAR GAMBAR

Gambar 2.1 <i>Parallel HEV</i>	8
Gambar 2.2 <i>Diagram Simulink Plant Parallel HEV</i>	9
Gambar 2.3 <i>Linier Model</i>	11
Gambar 2.4 <i>Model Kendaraan</i>	17
Gambar 2.5 <i>Model Roda Kendaraan</i>	19
Gambar 2.6 <i>Struktur Disturbance Observer</i>	22
Gambar 2.7 <i>Struktur Penyederhanaan Disturbance Observer</i>	22
Gambar 2.8 <i>Struktur Fuzzy Controller</i>	24
Gambar 2.9 <i>Struktur Fuzzy Controller pada Sistem Closed Loop</i>	25
Gambar 2.10 <i>Sistem Inferensi Fuzzy TSK</i>	26
Gambar 2.11 <i>Struktur Neuro-Fuzzy</i>	26
Gambar 3.1 <i>Komponen Gaya Kendaraan saat Melaju</i>	30
Gambar 3.2 <i>Sistem Kontrol PHEV dengan Fuzzy Controller Tipe Mamdani</i>	34
Gambar 3.3 <i>Diagram Simulink Fuzzy Controller</i>	34
Gambar 3.4 <i>Membership Function Input</i>	36
Gambar 3.5 <i>Membership Function Output</i>	37
Gambar 3.6 <i>Diagram Bode Filter</i>	40
Gambar 3.7 <i>Diagram Simulink Pencarian Data Training ANFIS</i>	41
Gambar 3.8 <i>Membership Function Input ANFIS</i>	42
Gambar 3.9 <i>Blok Diagram Fuzzy Disturbance Observer</i>	43
Gambar 3.10 <i>Diagram Simulink Fuzzy Disturbance Observer</i>	44
Gambar 4.1 <i>Respon Sistem Pengujian tanpa Disturbance</i>	45
Gambar 4.2 <i>Respon Sistem Pengujian dengan Disturbance I</i>	47
Gambar 4.3 <i>Respon Sistem Pengujian dengan Disturbance II</i>	48
Gambar 4.4 <i>Respon Sistem Pengujian dengan Disturbance III</i>	49
Gambar 4.5 <i>Respon Sistem Pengujian dengan Disturbance IV</i>	51
Gambar 4.6 <i>Respon Sistem Pengujian dengan Disturbance V</i>	52
Gambar 4.7 <i>Respon Sistem Pengujian dengan Disturbance VI</i>	53

Fuzzy Disturbance Observer with Neuro-Fuzzy Invers Model for Parallel Hybrid Electric Vehicle Speed Controller

Yoga Alif Kurnia Utama ¹, Trihastuti Agustinah ², Josaphat Pramudijanto ³

Abstrak—Pada dekade terakhir, terdapat banyak sekali kendaraan transportasi yang menggunakan *Internal Combustion Engine* (ICE). Hal ini menimbulkan dampak serius pada lingkungan karena emisi gas polutan yang dikeluarkannya. Salah satu solusi adalah penggunaan *hybrid electric vehicle* (HEV) sebagai pengganti dari kendaraan yang menggunakan ICE. Salah satu performa yang harus dimiliki oleh HEV adalah memiliki kecepatan yang tetap saat melaju di atas jalan raya. Jalan raya yang tidak rata akan menyebabkan gangguan pada HEV sehingga kecepatannya menjadi tidak tetap. Pada penelitian ini, metode *fuzzy disturbance observer* akan digunakan untuk menolak gangguan yang diterima HEV yang dikarenakan oleh jalan yang tidak rata. *Neuro-fuzzy invers model* digunakan sebagai komponen *invers model* dalam *disturbance observer* dimana mempunyai kelebihan dalam menggambarkan *invers model* pada *plant* tanpa membuat model matematikanya. Pengujian dilakukan dengan membandingkan antara dua metode kontrol yaitu *fuzzy* dan *fuzzy disturbance observer* pada enam tipe model jalan yang berbeda. Hasil simulasi menunjukkan bahwa metode *fuzzy disturbance observer* dengan *neuro-fuzzy invers model* ini dapat menolak *disturbance* yang dikarenakan oleh jalan raya yang tidak rata dan dapat mempertahankan kecepatannya.

Kata Kunci—*Parallel Hybrid Electric Vehicle, Fuzzy Control, Disturbance Observer, Neuro-Fuzzy Invers Model*

Abstract—In the last decade, there are a lot of transportation vehicles that use Internal Combustion Engine (ICE). They make serious problem to natural environment that caused by gas emissions pollutant which produces from them. One of many solutions is using Hybrid Electric Vehicle (HEV) as replacement of ICE system in the vehicles. One of the performance that must be have in HEV is having a steady speed while driving down on the road. Bumpy road would cause a disturbance in HEV so that the speed of HEV will not steady. In this research, fuzzy disturbance observer method use to reject the disturbance of HEV that caused by bumpy road. Neuro-fuzzy inverse model use as a component of the inverse model in disturbance observer which has advantages that can describe the inverse model without make a mathematic model. Testing will compare between two controllers, those are fuzzy and fuzzy disturbance observer in six kind of road models. The simulation results show that the method of fuzzy disturbance observer with neuro-fuzzy inverse model can reject the disturbance that caused by bumpy road and maintain its speed.

Keyword—*Parallel Hybrid Electric Vehicle, Fuzzy Control, Disturbance Observer, Neuro-Fuzzy Invers Model*

I. INTRODUCTION

At the present time, the land transportation is crowded with many conventional fossil-using vehicles. These kind of vehicles use Internal Combustion Engine (ICE) as the driving force. One of the weakness of ICE vehicles is the pollutant emission. Those pollutant is one of many causes that responsible to global warming issue nowadays.

The uses of hybrid electric vehicle (HEV) is one of the solution to overcome the pollutant emission problem. HEV is a certain vehicle type that has two source of drive power; ICE and electric motor. The source of energy is from the gasoline fuel, and the electricity potentation saved in the battery. With the presence of this

hybrid engine, lower hazardous emission is expected, thus the polution it makes can be much lower than the conventional vehicle does.

One of the performances HEV must have is the resistance of any disturbances it may get on the road. The disturbances can come from various things: bumpy road, wind, etc. The disturbances make the speed of vehicle itself becomes unstable. In order of that, those kind of disturbances must be overcome so then the speed of vehicle becomes stable. Some of the researchers have made researchs of these speed controller of the HEV.

In [1], the research use fuzzy controller in a speed simulation of HEV, which uses reference speed alteration. The research's result is the actual speed follows the reference speed as long as the load is not altered. The load makes the actual speed differ from the reference speed.

In [2], regarding the tracking speed on parallel HEV with fuzzy controller, input shifting reference speed was used in the test. This research shows that fuzzy controller can make actual speed track the reference speed. However if the load is changed, declining actual speed occurs. This weaknesses must be solved, so a certain strategy is needed to overcome the shifting load on parallel HEV.

The shifting loads can be detected using disturbance observer [3]. The results show that disturbance observer

Yoga Alif Kurnia Utama is with Departement of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia. E-mail: yoga.lehrer1@gmail.com

Trihastuti Agustinah is with Departement of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia. E-mail: trihastuti@ieec.org

Josaphat Pramudijanto is with Departement of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia. E-mail: pramudijanto@gmail.com

can compensate the shift in loads, frictions, and the occurred modelling error. Because in the disturbance observer is posed invers model component, thus a certain method is needed to create certain plant's invers model.

In another research [4], it is said that using neuro-fuzzy invers model can be used to describe a plant invers model without creating its mathematical model. In that research is shown that neuro-fuzzy invers model has smaller mean square error compared to that invers model with parametric model.

Thus, in order that HEV can follow the reference speed, and can resist any disturbances, we have idea to make fuzzy controller that is combined with disturbance observer, where the invers model uses the neuro-fuzzy invers model.

II. METHOD

A. Plant Parallel HEV Model

The plant parallel HEV model is shown on Figure 1[5].

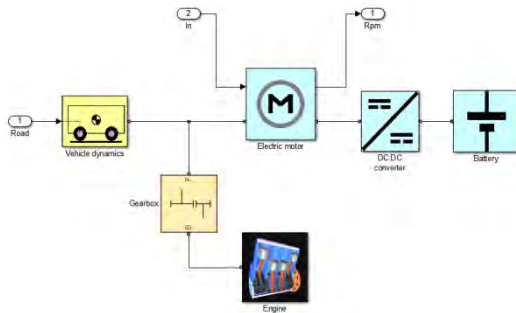


Figure 1. Plant Parallel HEV Simulink Diagram

In this research, the controller is going to be applied to a plant parallel HEV model. This plant model is obtained from simulink on MATLAB R2014a.

In the diagram above, there are six blocks of the main parts. Six blocks are consisted of engine, electric motor, gearbox, DC-DC converter, battery, and vehicle dynamics. Engine and electric motor is used as a driver to parallel HEV, which the engine as main driver, and electric motor as secondary driver of parallel HEV. Both of them have different axis, thus gearbox block is needed to combine the two of those axial driver. The electric motor gets its power through battery, where the voltage is set by DC-DC converter. Both parallel HEV motor (engine and electric motor) will together drive the parallel HEV as a unity of a block named vehicle dynamic.

B. Design of Disturbance

In this research uses a type of disturbance, which comes from the road crossed by PHEV model, which is tilted to certain degree. the Disturbance produced from the tiltness of the road is chosen because it affects directly to the vehicle's velocity. Descending road tends to make vehicle move faster. This happens because projection of weight pushes down the vehicle, thus the velocity is increased. To overcome this issue, the rotation of the HEV machine must be diminished to keep the velocity still. Vice versa, ascending road tends to make vehicle slower, because weight projection pulled down the vehicle.

Thus the velocity will be slower. The explanation can be seen in Figure 2. Figure 2 shows that weight force

component $mg\sin\beta$ which affects the vehicle's speed. Weight force component $mg\sin\beta$ will produce disturbance torque to the wheels. How big the disturbance depends on the distance between wheel rotating point, to perpendicularly crosses the working force line $mg\sin\beta$, which is notated as $h-r$.

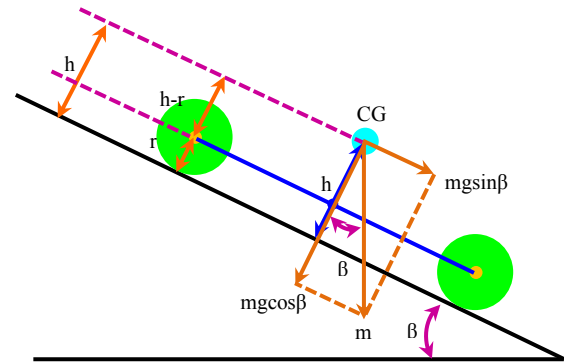


Figure 2. Parallel HEV's Force Component

So then the disturbance torque can be depicted in (1).

$$\tau_{dis} = mg\sin\beta(h - r) \quad (1)$$

τ_{dis} = disturbance torque (Nm)

m = vehicle's mass (kg)

g = gravity force (m/s^2)

β = angle of road ($^\circ$)

h = distance between center of gravity and ground level(m)

r = wheel radius (m)

Using the parameter value of $m = 1200$ kg, $g = 9.81$ m/s^2 , $h=0.8$ m, $r = 0.3$ m dan $\beta = 20^\circ$ for descending road and $\beta = -20^\circ$ for ascending road then the torque for the descending road is:

$$\tau_{dis} = 1200 \cdot 9.81 \cdot \sin(20)(0.8 - 0.3) = 2013 \text{ Nm}$$

While the torque for ascending road is:

$$\tau_{dis} = 1200 \cdot 9.81 \cdot \sin(-20)(0.8 - 0.3) = -2013 \text{ Nm}$$

Table 1. Road Model

No	Road Model	No	Road Model
1		4	
2		5	
3		6	

In this simulation, the disturbances gotten from the tilt road is divided into six type. These six type are observable in Table 1.

C. Design Fuzzy Disturbance Observer

Fuzzy disturbance observer is the combination of fuzzy controller dan disturbance observer. First, fuzzy controller is made using mamdani type, which is used to control the speed of electric motor. The fuzzy controller has 2 membership function inputs error of angular motor velocity (e), shifting error of angular motor velocity (Δe), and 1 membership function output that is control signal (u). Membership function is shown in Figure 3.

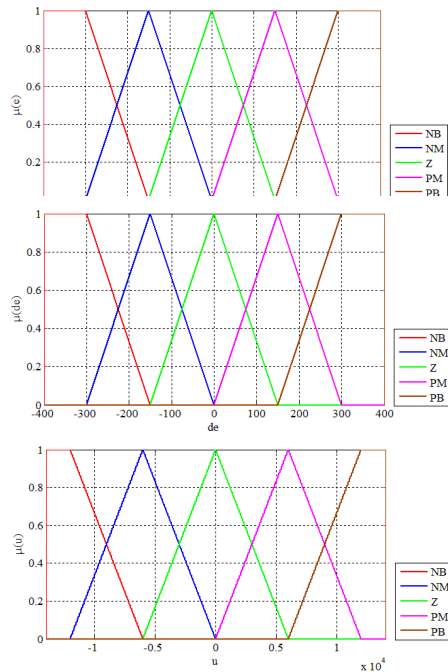


Figure 3. Membership Function Fuzzy Controller

Rule base used in this fuzzy controller is shown in Table 2.

Table 2. Rule Base Fuzzy Controller

e \ de	NB	NM	Z	PM	PB
NB	NB	NB	NB	NM	Z
NM	NB	NB	NM	Z	PM
Z	NB	NM	Z	PM	PB
PM	NM	Z	PM	PB	PB
PB	Z	PM	PB	PB	PB

Disturbance observer has two components, they are low pass filter and invers model. Low pass filter design follow (2).

$$Q(s) = \frac{1 + \sum_{k=1}^{N-r} a_k(\tau s)^k}{1 + \sum_{k=1}^N a_k(\tau s)^k} \quad (2)$$

N = $Q(s)$ order

r = $Q(s)$ relative's order

τ = cutoff frequency

a_k = $Q(s)$ coefficient

The higher the order of the filter, the bigger filter bandwidth needed to repel the disturbances. However, with higher filter order, the phase will fall behind. It

must be strictly considered so the filter must have order as smallest as possible, thus the phase will not fall behind. Because the goal of this research is to create plant parallel HEV that can quickly overcome disturbances, so filter with one order is made.

Disturbance that happens is assumed to not exceed frequency of 2 rad/s, so the filter will be made with frequency cut-off as high as 2 rad/s. By referring to (2) and by choosing filter coefficients that appropriate with Butterworth coefficients, so the filter equation for one order can be made as seen (3).

$$Q(s) = \frac{1 + \sum_{k=1}^{N-r} a_k(\tau s)^k}{1 + \sum_{k=1}^N a_k(\tau s)^k}$$

$$Q(s) = \frac{1 + \sum_{k=1}^{1-1} a_k(\tau s)^k}{1 + \sum_{k=1}^1 a_k(\tau s)^k}$$

$$Q(s) = \frac{1}{1 + a_k \tau s} \quad (3)$$

Since the *Butterworth coefficient* in one order is $s+1$ so the value of $a_k=1$. Previously it has been decided that cutoff frequency used 2 rad/s. So the filter which is going to be used in the *disturbance observer* is shown in (4) below.

$$Q(s) = \frac{1}{1 + a_k \tau s}$$

$$Q(s) = \frac{1}{1 + 1 \tau s}$$

$$Q(s) = \frac{1}{1 + \tau s} = \frac{1}{1 + \frac{1}{\omega_c} s}$$

$$Q(s) = \frac{1}{1 + \frac{1}{2} s} = \frac{1}{1 + 0.5s} \quad (4)$$

The invers model in disturbance observer uses neuro-fuzzy invers model. The making of this neuro-fuzzy needs pairs of input-output data which are used in neuro-fuzzy network training. The form of input datas are angular velocity of electric motor (ω) while output datas are control signals (u).

The neuro-fuzzy training uses toolbox in MATLAB 2014a, where the neuro-fuzzy network contains six membership function inputs with generalized bell-shaped type and six membership function output with linear type. This Training is done in order to obtain premis parameter, and consequent parameters.

From the result of the training, premise parameters result are obtained for all six membership function with generalized bell-shaped type as observable in Table 3.

Table 3 Premise Parameters

Parameter MF	a	b	c
1	32.86	3.451	0.4372
2	36.38	3.367	68.22
3	32.17	0.3371	123.3
4	33.23	2.619	194.7
5	33.1	2.075	260
6	32.69	0.6021	325.4

The result of *consequent parameters* can be seen in Table 4.

Table 4. Consequent Parameters

MF \ Parameter	c_1	c_2
1	1.11	-128
2	0.9031	-140.3
3	-2.638	190.4
4	0.154	-51.81
5	0.05891	-27.96
6	-3.472	1253

From the disturbance and fuzzy disturbance observer design that have done before, so the result can be used for diagram block that can be seen in Figure 4.

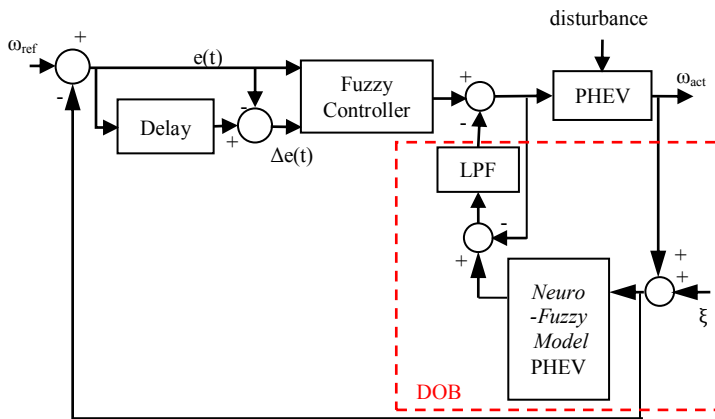


Figure 4. Fuzzy Disturbance Observer's Diagram Block

III. RESULT AND DISCUSSION

This controller will be tested by observing the endurance of the system against the various of disturbances, and observing the system's responses. The test uses two kind of controller, that is: fuzzy controller and fuzzy disturbance observer which uses neuro-fuzzy invers model. In this test, these two controller will be tested with six disturbances created from six different type of road, as shown in Table 1. It is done with reference speed of 10 m/s.

The test result produces the system responses as shown in Figure 5-10. These respons has 4 graphics in 1 figure. There are reference, fuzzy, fuzzy disturbance observer (fuzzy-DOB) and angle of road graphic. From these graphic, we can see which controller that have better respon and when HEV get the disturbance.

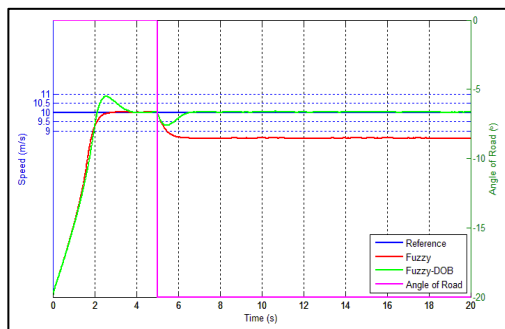


Figure 5. Testing with Disturbance I

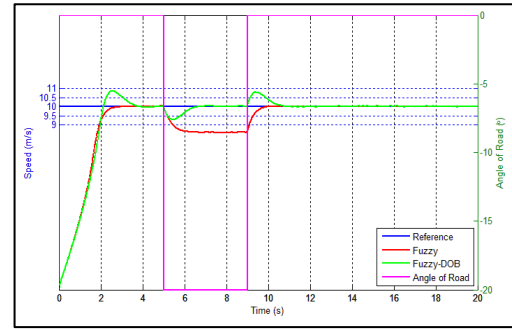


Figure 6. Testing with Disturbance II

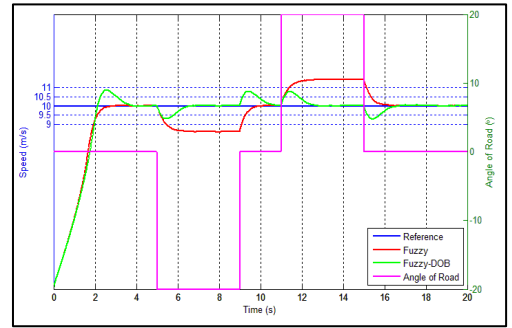


Figure 7. Testing with Disturbance III

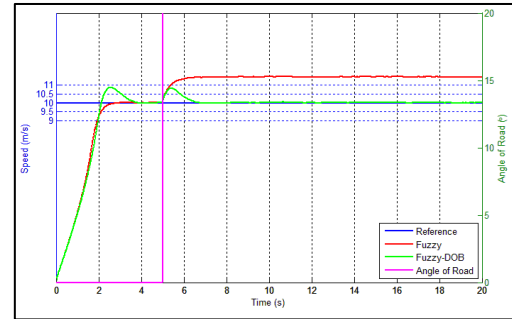


Figure 8. Testing with Disturbance IV

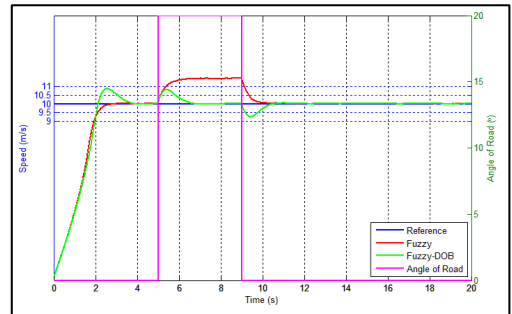


Figure 9. Testing with Disturbance V

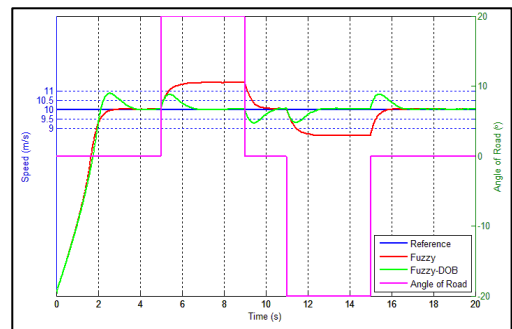


Figure 10. Testing with Disturbance VI

The result of simulation shows that fuzzy disturbance observer controller will make actual speed on par with the reference speed that 10 m/s. Whenever the PHEV experiences disturbances, the actual speed declines to 9.3 m/s if PHEV goes through ascending road, and going up to 10.8 m/s if PHEV goes through descending road. However, even though there is different between actual speed and reference speed, 1.75 seconds post-disturbance the actual speed will move 10 m/s.

Different simulation result is detected when fuzzy controller is used in PHEV model. Fuzzy controller will make actual speed decline to 8.6 m/s in ascending road, andgo up to 11.4 m/s in descending road. Fuzzy controller will make actual speed lead to reference speed only when PHEV does not experience any disturbances. So when PHEV experiences disturbances, the actual speed will not be same as reference speed.

Based on seven testing, it can be concluded in Table 5 which steady state error from each controllers.

Table 5. Steady State Error

Controller	Angle of Road		
	0°	20°	- 20°
Fuzzy	0.05 m/s	1.4 m/s	-1.4 m/s
Fuzzy Disturbance Observer	0.05 m/s	0.05 m/s	0.05 m/s

III. CONCLUSION

In this paper, two controller for PHEV speed controller are presented. They are fuzzy and fuzzy disturbance observer. Both of them applied to PHEV on the six different road model.

From this result, a conclusion can be drawn, that is fuzzy disturbance observer can be applied to plant parallel HEV where the controller can deflect disturbances produced from the different tilt angle of the roads.

REFERENCES

- [1] A.M. Harsha S. Abeykoon, Hasala R. Senevirathne, "Disturbance observer based current controller for a DC motor", Pages:47– 52, 2012.
- [2] J.C. Gonzalez-Gomez, J.A. Ruz-Hernandez, R. Garcia-Hernandez, and E.N. Sanchez, "Real-time *neuro-fuzzy* inverse control applied to a DC motor", *2011 8th International Conference on*, Pages:1– 5, 2011.
- [3] Mariam Khan and Narayan C.Kar, "Speed tracking performance of *fuzzy* based vector control of induction motor drives for hybrid electric vehicles", *Canadian Conference on*, Pages:000607– 000610, 5 May 2008.
- [4] S. Wahsh, H. G. Hamed, M. N. F. Nashed, and T. Dakroy, "Fuzzy logic based control strategy for *parallel* hybrid electric vehicle", *Proceedings of 2008 IEEE International Conference on Mechatronics and Automation*, Pages:27 – 31, 20 May 2008.
- [5] Mathworks Corporate Headquarters, "*Parallel Hybrid Transmission*", 12 Desember 2014, <http://www.mathworks.com/help/phymod/sdl/examples/parallel-hybrid-transmission.html>.

Design Fuzzy Disturbance Observer with Neuro-Fuzzy Inverse Model for Parallel Hybrid Electric Vehicle Speed Controller

**Yoga Alif Kurnia Utama
Dr. Trihastuti Agustinah, S.T., M.T.
Ir. Josaphat Pramudijanto, M.Eng.**

**Control System Engineering
Faculty of Industrial Engineering
Institut of Technology Sepuluh Nopember
2015**

Background

- * Using of fuzzy controller (FC) for speed controller of the parallel hybrid electric vehicle (PHEV) [1][2].
- * Speed decrease when the load increase [1][2].

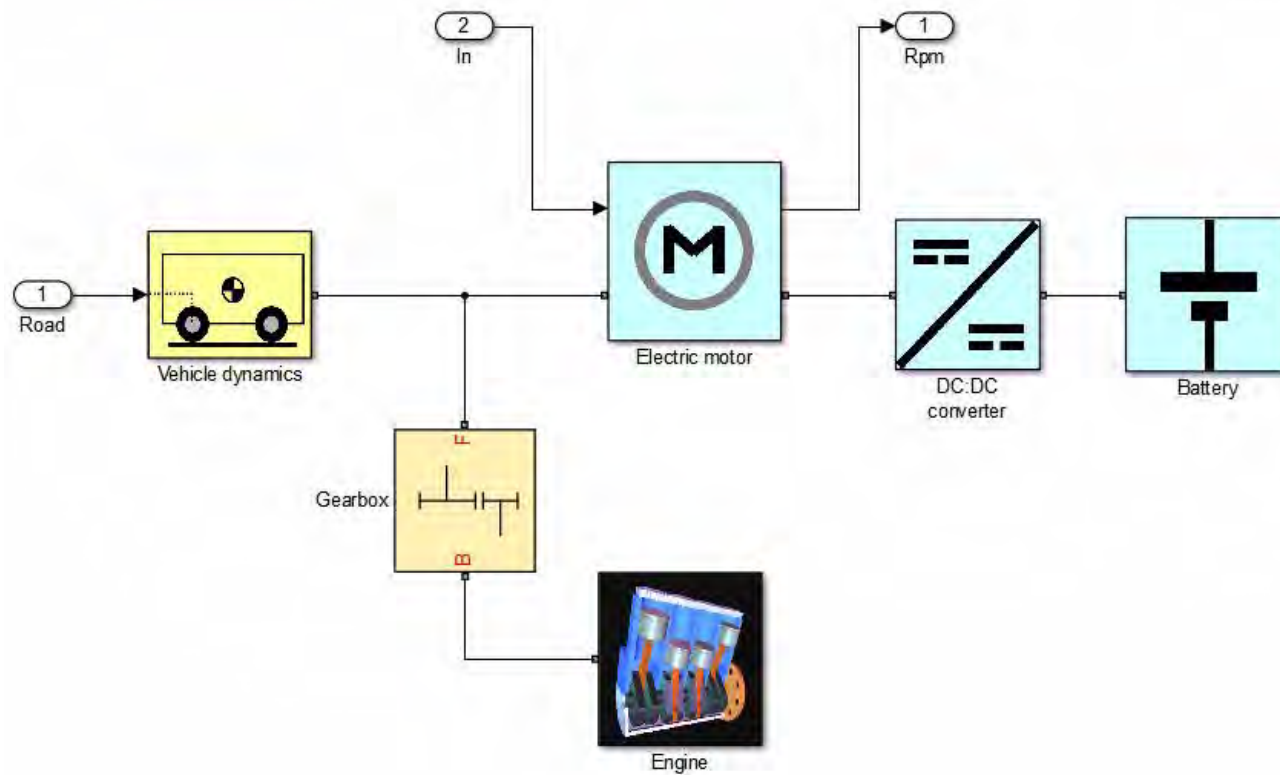
Problem Statement

- * FC can not maintain speed of HEV when the load changes [1] [2].
- * Disturbance Observer (DOB) can compensate for load variation, friction, and modeling errors in a plant [3].
- * Neuro-fuzzy inverse model can be used in plant without creating its mathematical model [4].

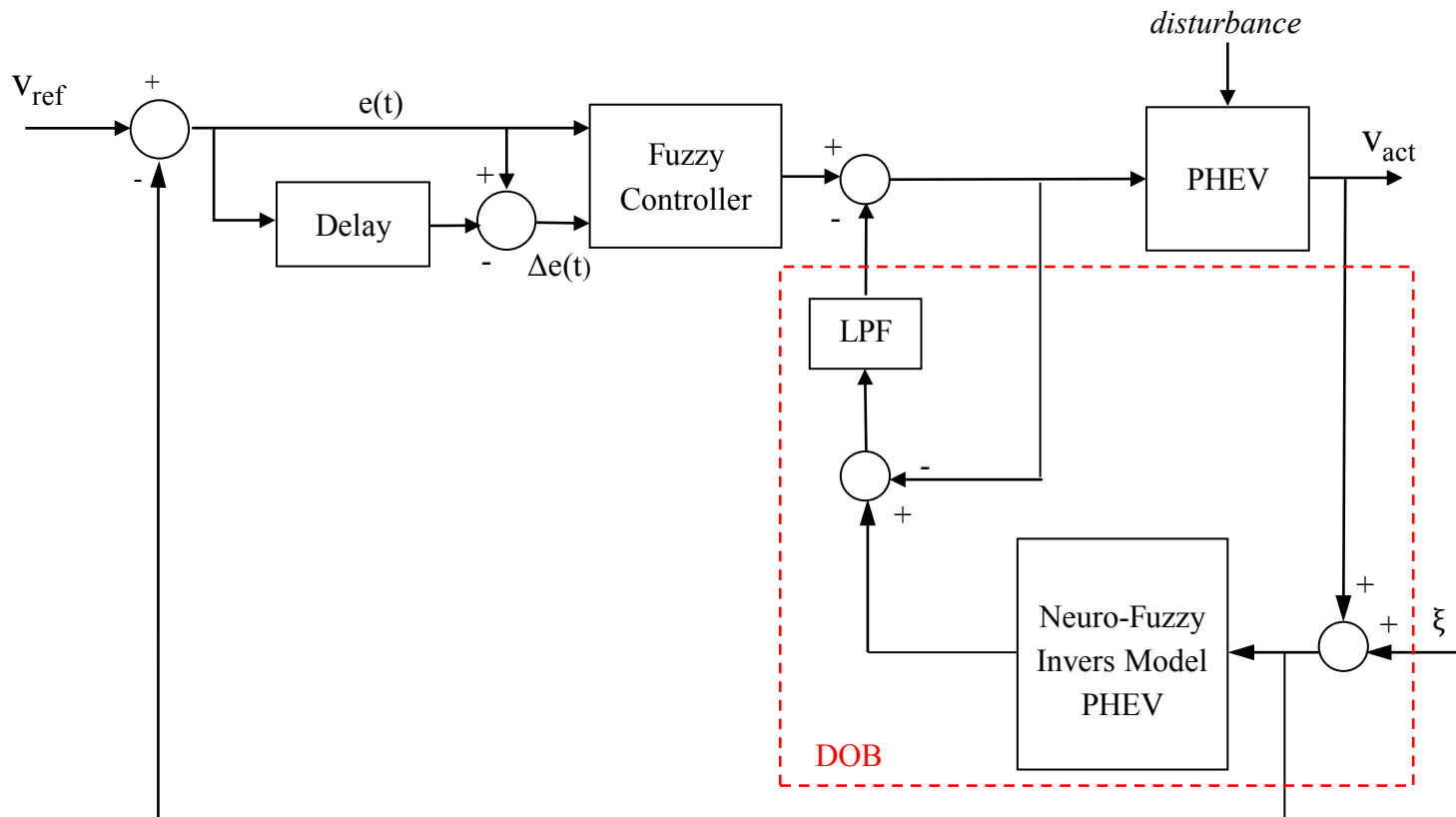
Objective

Fix the weakness of FC by adding DOB where the inverse model of DOB is neuro-fuzzy inverse model so PHEV will be resistant from load variation (disturbance).

PHEV Model



Block Diagram



Disturbance Observer

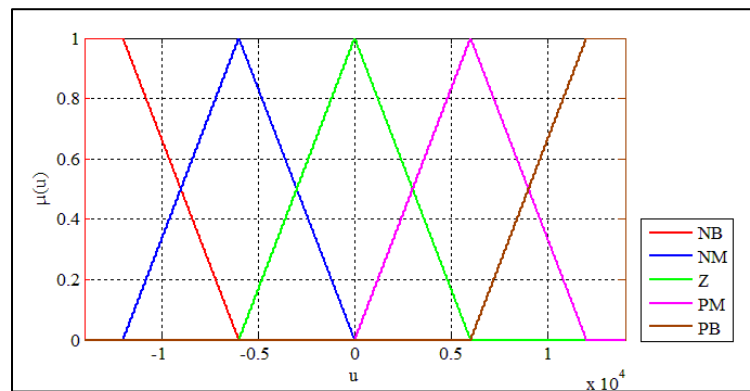
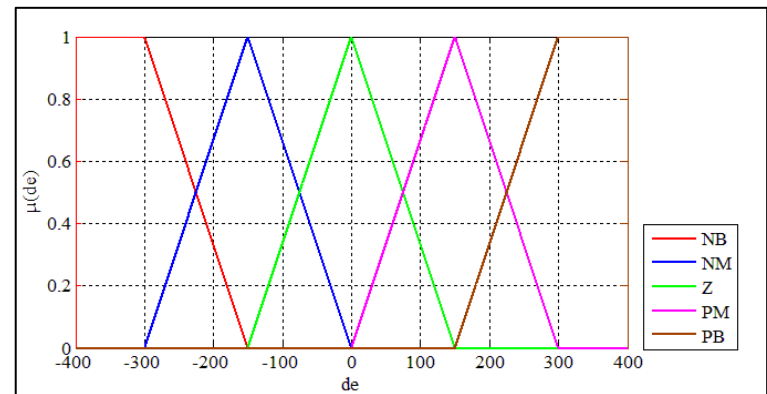
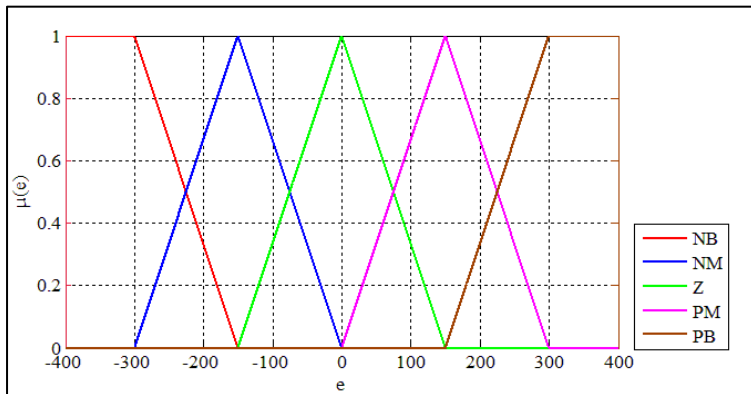
$$Q(s) = \frac{1}{1 + a_k \tau s}$$

$$Q(s) = \frac{1}{1 + \tau s}$$

$$Q(s) = \frac{1}{1 + \frac{1}{\omega_c} s}$$

$$Q(s) = \frac{1}{1 + \frac{1}{2} s} = \frac{1}{1 + 0.5s}$$

Membership Function



Rule Base

<div>de</div> <div>e</div>	NB	NM	Z	PM	PB
NB	NB	NB	NB	NM	Z
NM	NB	NB	NM	Z	PM
Z	NB	NM	Z	PM	PB
PM	NM	Z	PM	PB	PB
PB	Z	PM	PB	PB	PB

Neuro-Fuzzy

Parameter MF	a	b	c
1	32.86	3.451	0.4372
2	36.38	3.367	68.22
3	32.17	0.3371	123.3
4	33.23	2.619	194.7
5	33.1	2.075	260
6	32.69	0.6021	325.4

Input:

6 Generalized Bell-Shaped MF

$$\mu(\omega) = \frac{1}{1 + \left| \frac{\omega - c}{a} \right|^{2b}}$$

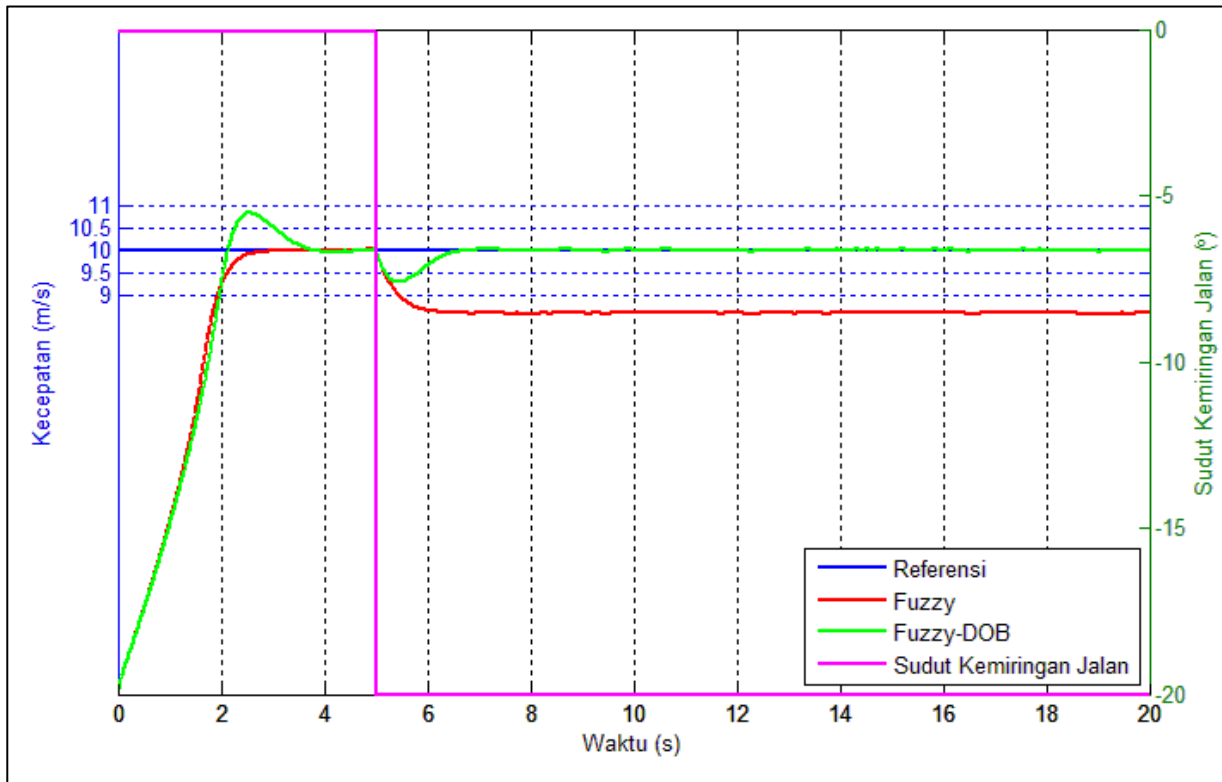
Parameter MF	c ₁	c ₂
1	1.11	-128
2	0.9031	-140.3
3	-2.638	190.4
4	0.154	-51.81
5	0.05891	-27.96
6	-3.472	1253

Output:

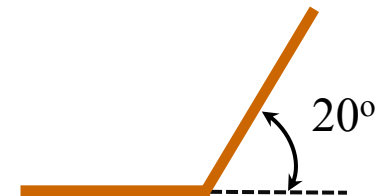
6 Linier MF

$$u = c_1 \omega + c_2$$

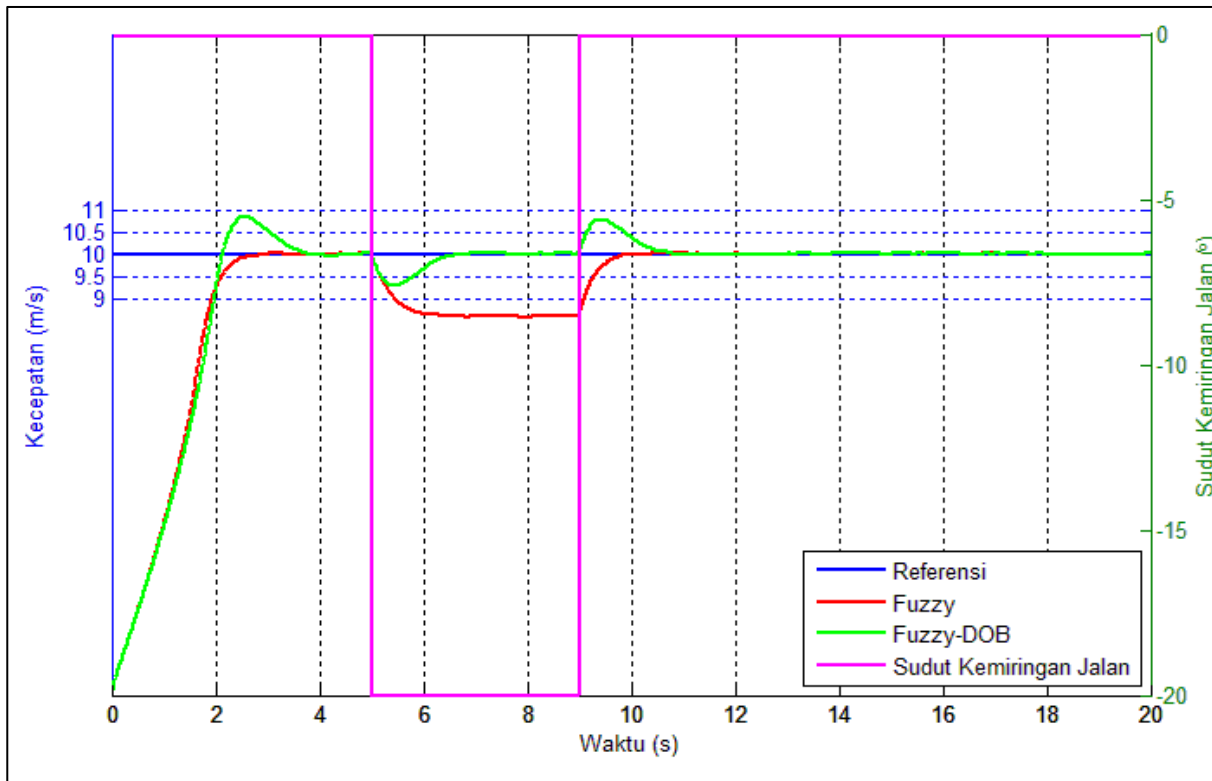
Testing with Disturbance I



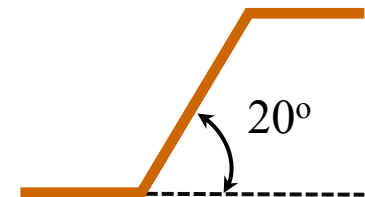
Road Model:



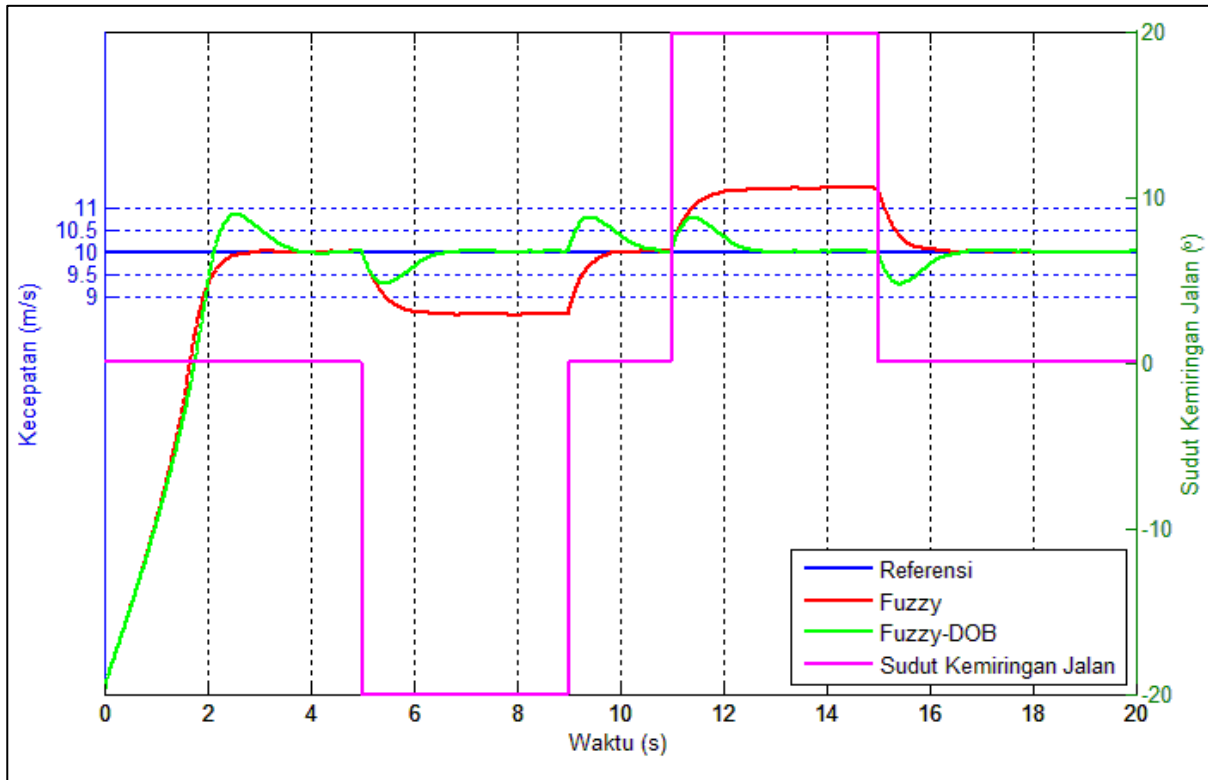
Testing with Disturbance II



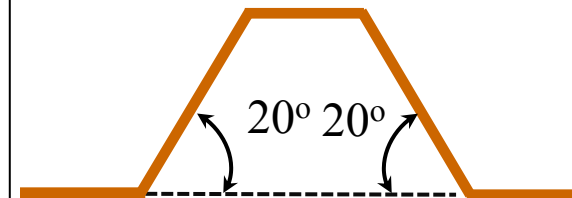
Road Model:



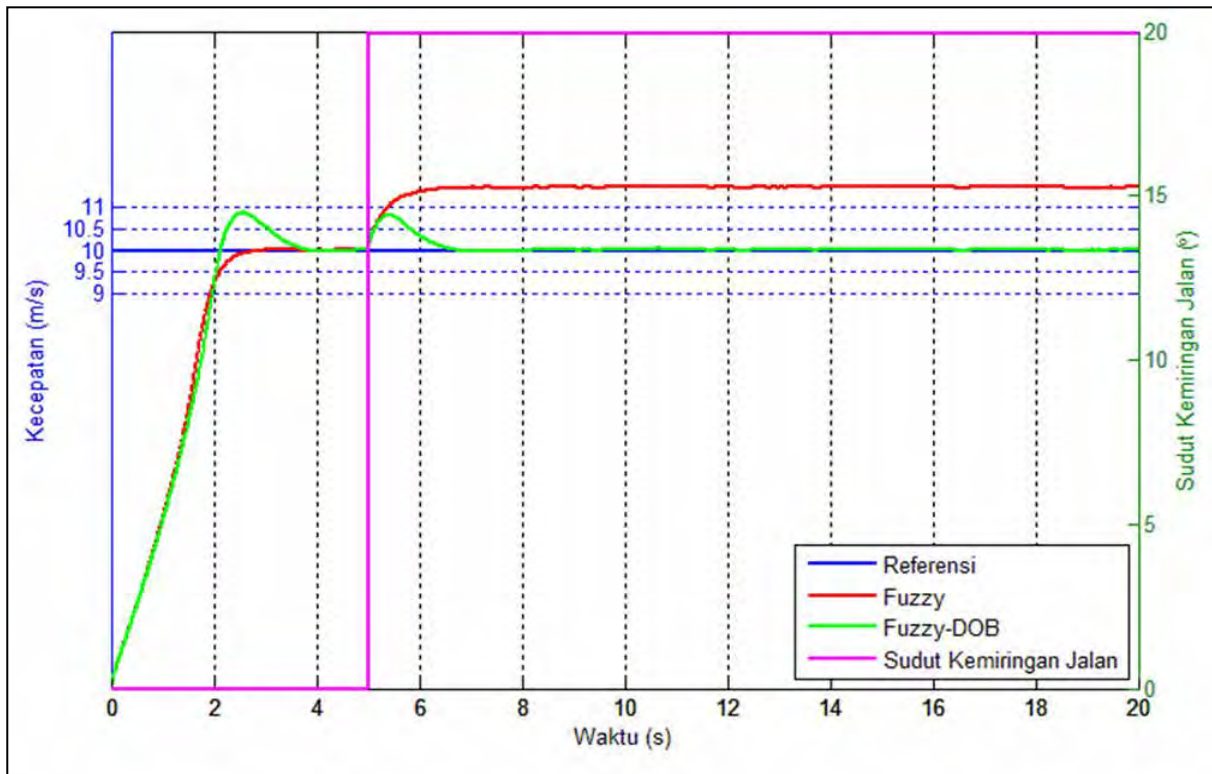
Testing with Disturbance III



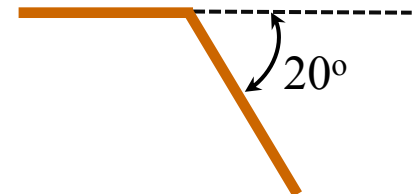
Road Model:



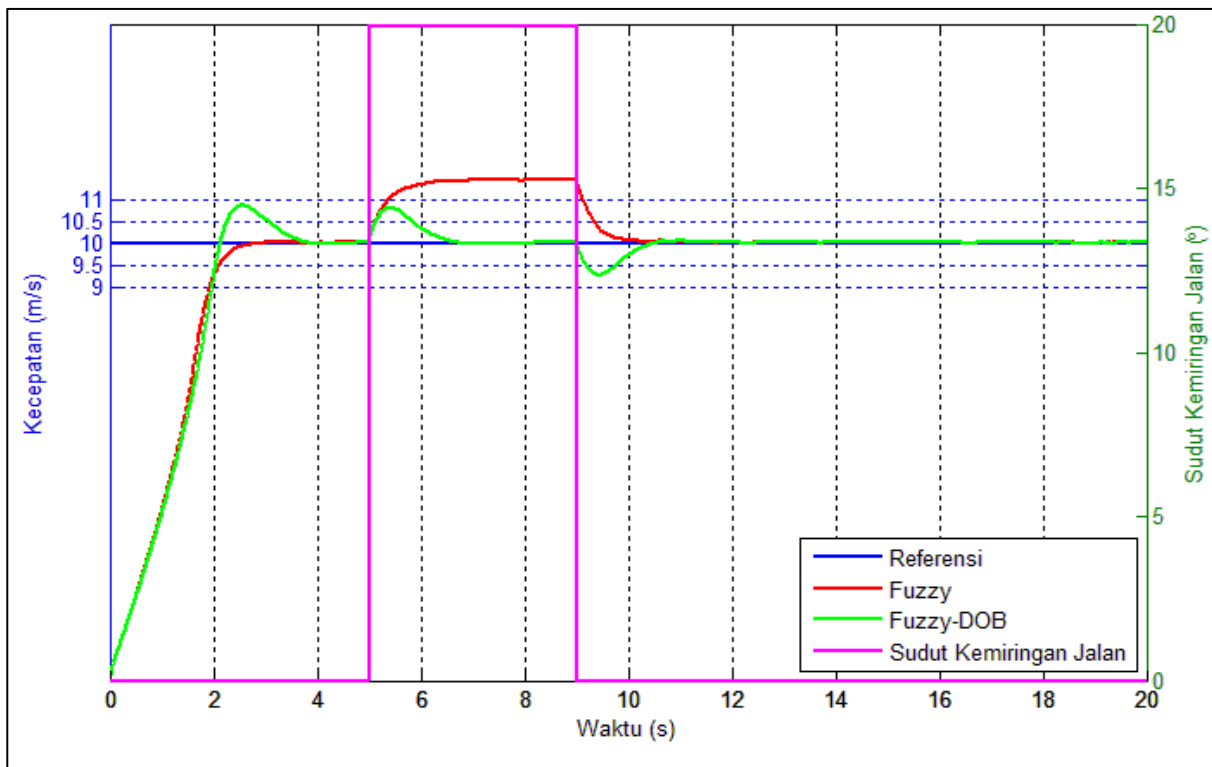
Testing with Disturbance IV



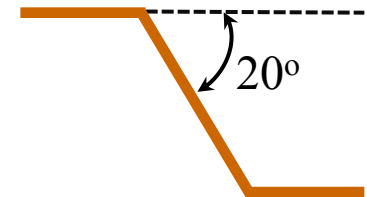
Road Model:



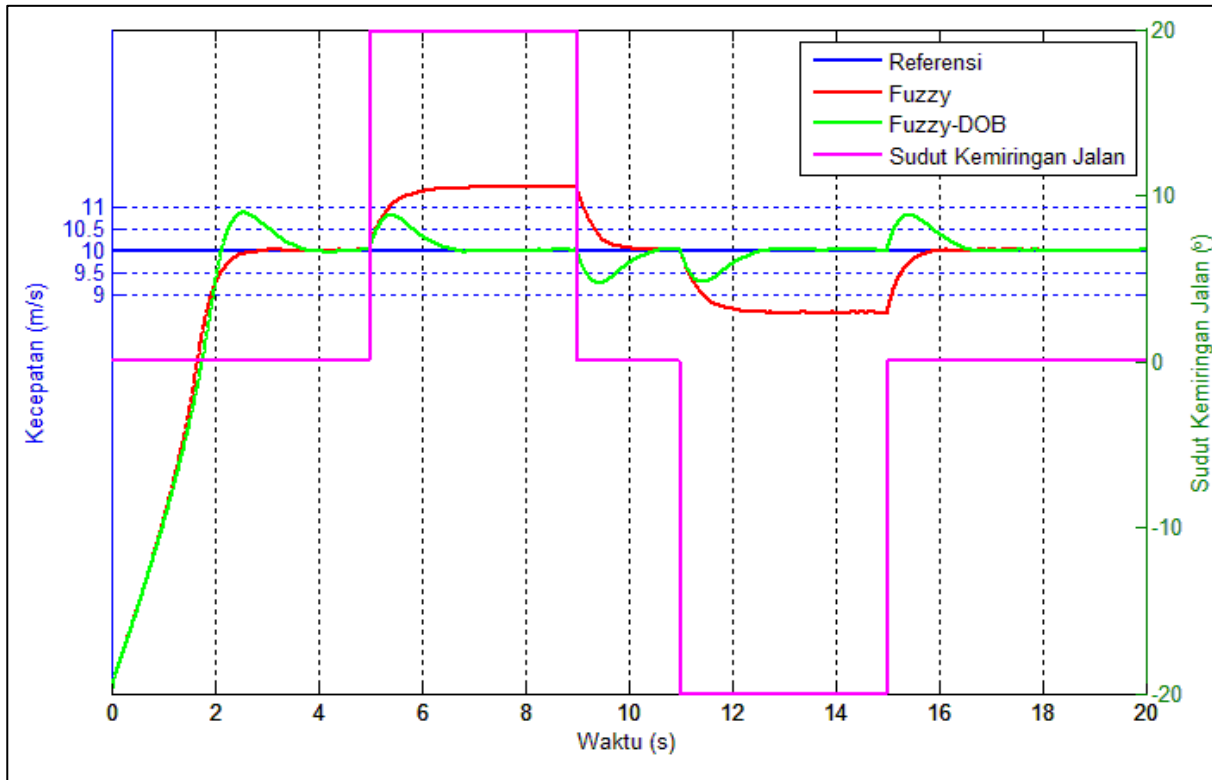
Testing with Disturbance V



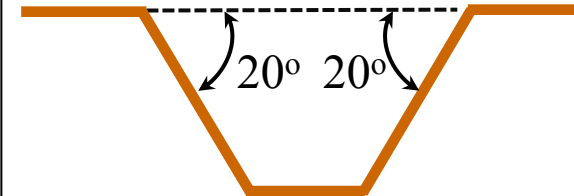
Road Model:



Testing with Disturbance VI



Road Model:



CONCLUSION

Controller	Angle of Road		
	0°	20°	- 20°
Fuzzy	0.05 m/s	1.4 m/s	-1.4 m/s
Fuzzy Disturbance Observer	0.05 m/s	0.05 m/s	0.05 m/s

References

- [1] S. Wahsh, H. G. Hamed, M. N. F. Nashed, and T. Dakroy, "Fuzzy Logic Based Control Strategy for Parallel Hybrid Electric Vehicle", *Proceedings of 2008 IEEE International Conference on Mechatronics and Automation*, Pages:27 – 31, 20 May 2008.
- [2] Mariam Khan and Narayan C.Kar, "Speed Tracking Performance of Fuzzy Based Vector Control of Induction Motor Drives for Hybrid Electric Vehicles", *Canadian Conference on Electrical and Computer Engineering*, Pages:000607– 000610, 5 May 2008.
- [3] A.M. Harsha S. Abeykoon, Hasala R. Senevirathne, "Disturbance Observer Based Current Controller for a DC motor", *International Conference on Information and Automation for Sustainability*, Pages:47– 52, 2012.
- [4] J.C. Gonzalez-Gomez, J.A. Ruz-Hernandez, R. Garcia-Hernandez, and E.N. Sanchez, "Real-Time Neuro-Fuzzy Inverse Control Applied to A DC Motor", *Western Athletic Conference*, Pages:1– 5, 2011.
- [5] Mathworks Corporate Headquarters, "Parallel Hybrid Transmission, 12 Desember 2014, <http://www.mathworks.com/help/physmod/sdl/examples/parallel-hybrid-transmission.html>.



**THANK YOU
FOR
YOUR ATTENTION**